



Grado en Ingeniería Informática

<Ingeniería del Software>

Trabajo Fin de Grado

MoviesHD Portal web para la venta de películas

Autor: Sergio López Díaz

Tutor: Javier Sánchez Pérez

Resumen

El presente trabajo de grado consiste en el desarrollo e implementación de un Portal Web para la venta de películas en formato digital. La principal característica de este proyecto es el proceso de pago, donde un usuario se registra en el sitio web y puede realizar la compra de películas para su futura visualización.

La metodología utilizada en el desarrollo del proyecto ha sido SCRUM, con ella se ha dividido el proceso de desarrollo en la creación inicial del Product Backlog (historias de usuarios) y cuatro Sprint (conjunto de historias de usuarios).

En el Sprint cero es donde se aprenden las nuevas herramientas y tecnologías a usar, el diseño de la base de datos y la integración con los Servicios Web de Amazon. En los restantes Sprint es donde se lleva a cabo todo el desarrollo del trabajo de fin de grado.

Abstract

The present work of degree consists in the development and implementation of a Web Portal for the sale of films in digital format. The main feature of this project is the purchase process, where a user registers on the website and can make the purchase of films for future viewing.

The methodology used in the development of the project has been SCRUM, with it the development process has been divided into the initial creation of the Product Backlog (user stories) and four Sprint (set of user stories).

Sprint zero is where you learn the new tools and technologies to use, the design of the database and the integration with Amazon Web Services. The rest of the Sprint is where all the development of the final degree project takes place.

AGRADECIMIENTOS

Escribo este apartado de agradecimientos para finalizar mi Trabajo de Fin de Grado. Ha sido un período de aprendizaje intenso, no solo en el campo de la informática, sino también a nivel personal. La realización de este proyecto ha tenido un gran impacto en mí y es por eso que me gustaría agradecer a todas aquellas personas que me han ayudado y apoyado durante este proceso.

Primero que todo, me gustaría agradecer a mis familiares, me habéis apoyado enormemente y siempre habéis estado ahí para ayudarme cuando lo necesitaba. Particularmente a mis padres y hermanos por sus sabios consejos y su comprensión.

Por otro lado, me gustaría darle las gracias a mi tutor Javier Sánchez Pérez, por su ayuda y sus consejos durante todo el transcurso del desarrollo del proyecto.

Finalmente, mis amistades más allegadas. No solo habéis estado ahí para apoyarnos entre nosotros en los momentos difíciles, sino que también hemos tenido charlas sobre otras cosas no relacionadas con universidades y proyectos de carrera.

¡Muchas gracias a todos!

Sergio López Díaz

ÍNDICE GENERAL

ın	aice (ie ngui	ras	XV
Ín	dice o	de tabla	as	XVI
1.	Intr	oducci	ón	1
	1.1.	Objeti	ivos iniciales	2
	1.2.	Comp	etencias específicas cubiertas	2
	1.3.	Aporta	aciones	3
2.	Esta	do acti	ual del tema	5
	2.1.	Platafo	ormas digitales	5
		2.1.1.	Netflix	5
		2.1.2.	HBO	6
		2.1.3.	Amazon Prime Video	
		2.1.4.	Filmin	8
		2.1.5.	MUBI	9
		2.1.6.	FlixOlé	10
		2.1.7.	Sky	11
	2.2.	Altern	nativas relacionadas con MoviesHD	12
		2.2.1.	Rakuten TV	12
		2.2.2.	Google Play Movies	13
		2.2.3.	Amazon	14
		2.2.4.	Microsoft Films & TV	15
	2.3.	Evalua	ación	16
3.	Plar	nificaci	ion del Portal Web	17
	3.1.	Planifi	dicación	17
	3.2.	Metod	lología	19
	3.3.	Presup	puesto	23
		3.3.1.	Costes de Personal	23
		3.3.2.	Costes de Hardware	23
		3 3 3	Costes de Software	23

XII ÍNDICE GENERAL

		3.3.4.	Otros costes
		3.3.5.	Costes totales
4.	Desa	arrollo	del Portal Web 25
	4.1.	Sprint	0
		4.1.1.	Aprendizaje y familiarización con las nuevas herramientas y tecnologías 25
		4.1.2.	Diseño de la base de datos
		4.1.3.	Integración inicial con los servicios de AWS
	4.2.	Sprint	1
		4.2.1.	HU 1. Página de inicio
		4.2.2.	HU 9. Detalle de la película
		4.2.3.	HU 6. Buscador de películas
		4.2.4.	HU 7. Filtro de búsqueda de películas
		4.2.5.	HU 2. Sistema de inicio de sesión tradicional
		4.2.6.	HU 5. Sistema de cierre de sesión
		4.2.7.	HU 4. Sistema de registro
		4.2.8.	Retrospectiva del Sprint
		4.2.9.	Mejoras del Sprint
		4.2.10.	Configuraciones extras del proyecto
	4.3.	Sprint	2
		4.3.1.	HU 12. Carro de la compra
		4.3.2.	HU 13. Detalle carro de la compra
		4.3.3.	HU 14. Pasarela de pago
		4.3.4.	HU 15. Confirmación de compra
		4.3.5.	HU 19. Acceso películas compradas
		4.3.6.	HU 18. Historial de compras
		4.3.7.	HU 17. Recuperar contraseña
		4.3.8.	Retrospectiva del Sprint
		4.3.9.	Mejoras del Sprint
	4.4.	Sprint	3
		4.4.1.	HU 3. Sistema de inicio de sesión con Google o Facebook
		4.4.2.	HU 10. Lista de favoritos
		4.4.3.	HU 11. Detalle de lista de favoritos
		4.4.4.	HU 16. Editar datos personales
		4.4.5.	HU 8. Paginación
		4.4.6.	HU 21. Preguntas frecuentes
		4.4.7.	HU 22. Contacto
		4.4.8.	HU 23. Política de datos personales
		4.4.9.	HU 24. Sobre nosotros

XIII

Bibliog	grafía	79
A.2.	Leyes de derecho de autor	77
A.1.	Leyes de protección de datos	75
A. Nor	mativa y legislación	75
5.2.	Trabajo futuro	74
5.1.	Conclusiones	73
5. Con	nclusiones y trabajo futuro	73
	4.4.13. Mejoras del Sprint	72
	4.4.12. Retrospectiva del Sprint	72
	4.4.11. HU 20. Multi-idioma	70
	4.4.10. HU 25. Términos y condiciones	70

ÍNDICE DE FIGURAS

2.1.	Netflix	5
2.2.	HBO	6
2.3.	Amazon Prime Video	7
2.4.	Filmin	8
2.5.	MUBI	9
2.6.	FlixOlé	10
2.7.	Sky	11
2.8.	Rakuten TV	12
2.9.	Google Play Movies	13
2.10.	Amazon	14
2.11.	Microsoft Films & TV	15
2.1	Madela Itanativa a Inggamental	17
3.1.	Modelo Iterativo e Incremental	
3.2.	Proceso de SCRUM	20
4.1.	Historias de usuario del Sprint 1	28
4.2.	Cabecera y carrusel de la página de inicio	29
4.3.	Secciones de la página de inicio	30
4.4.	Detalle de la película	32
4.5.	Buscador de película para dispositivos grandes	33
4.6.	Buscador de película para dispositivos pequeños y medianos	34
4.7.	Resultado de búsqueda de películas	34
4.8.	Filtro de búsqueda por categoría	35
4.9.	Filtro de búsqueda por idioma	35
4.10.	Formulario inicio sesión tradicional	36
4.11.	Formulario cierre de sesión	38
4.12.	Formulario de registro	39
4.13.	Historias de usuario del Sprint 2	40
4.14.	Notificación de añadir película al carro de la compra	41
4.15.	Detalle del carro de la compra	42
4 16	Pasarela de pago, proceso de selección del método de pago	44

XVI ÍNDICE DE FIGURAS

4.17. Pasarela de pago, compra mediante tarjeta	44
4.18. Pasarela de pago, compra mediante PayPal	45
4.19. Pasarela de pago, compra con la cuenta de PayPal	45
4.20. Pasarela de pago, aceptación de condiciones de pago con PayPal	46
4.21. Pasarela de pago, realización de compra mediante PayPal	46
4.22. Pasarela de pago, detalle de la compra mediante PayPal	47
4.23. Confirmación por correo electrónico de la compra	48
4.24. Archivo WEBVTT para el uso de las imágenes en miniaturas	49
4.25. Sprite, conjunto de imágenes en miniaturas	50
4.26. Listado de películas compradas	50
4.27. Reproductor multimedia Plyr	50
4.28. Listado de compra	53
4.29. Detalle de la compra	53
4.30. Factura de la compra en formato pdf	54
4.31. Formulario de envío del correo electrónico de recuperación de contraseña	56
4.32. Correo electrónico del proceso de recuperación contraseña	56
4.33. Formulario para el cambio de contraseña	57
4.34. Correo electrónico de confirmación del cambio de contraseña	57
4.35. Historias de usuario del Sprint 3	59
4.36. Formulario de inicio sesión con Google o Facebook	60
4.37. Inicio de sesión con Google	61
4.38. Inicio de sesión con Facebook	61
4.39. Películas favoritas	62
4.40. Listado de películas favoritas	63
4.41. Formularios para editar los datos personales y de acceso	65
4.42. Paginación	66
4.43. Pie de página de la web	67
4.44. Página de preguntas frecuentes	67
4.45. Página de contacto	69
4.46. Menú de opciones de idioma de la web	71

ÍNDICE DE TABLAS

3.1.	Tabla que refleja las distintas fases del desarrollo del proyecto	19
3.2.	Tabla que refleja las historias de usuarios del Product Backlog	22
3.3.	Tabla que refleja los costes mensuales del personal	23
3.4.	Tabla que refleja los costes hardware del proyecto	23
3.5.	Tabla que refleja otros costes mensuales del proyecto	24
3.6.	Tabla que refleja los costes totales del proyecto durante el período de un año	24

CAPÍTULO 1

INTRODUCCIÓN

En la actualidad, el entretenimiento digital es una de las industrias más crecientes y rentables. Son millones los usuarios que hacen uso plataformas digitales para el consumo cinematográfico. En España solamente, uno de cada tres hogares con acceso a Internet usa plataformas digitales de pago para ver contenidos audiovisuales online, según el último Panel de hogares de la Comisión Nacional de los Mercados y la Competencia (CNMC), publicado en noviembre del 2018.

La motivación en el tema del Trabajo de Fin de Grado (TFG) es el saber cómo se desarrolla una plataforma de streaming parecida a las existentes en el mercado actual y a la creación de un comercio electrónico relacionado con contenidos cinematográficos que pueda ser consumido desde un dispositivo móvil, tableta u ordenador.

El TFG MoviesHD consiste en un prototipo de aplicación web donde el usuario dispondrá de una gran variedad de películas, las cuales podrá comprar o añadir a su lista de deseos o favoritos, para ello, deberá estar registrado previamente. El usuario que no esté registrado podrá ver el catálogo completo de películas, buscar las películas por su nombre, filtrar la búsqueda por género e idioma. En cuanto al método de pago se hará uso de la API de Braintree, donde el usuario podrá realizar la compra de películas, ya sea a través de una cuenta PayPal o una tarjeta de crédito o débito. Los usuarios que no tengan una cuenta de PayPal podrán crearse una. Cuando un usuario realice una compra, se le enviará un correo electrónico con el detalle de dicha operación. La aplicación aparte de tener un formulario de registro, permitirá al usuario iniciar sesión a través de plataformas como Google o Facebook. En cuanto al usuario registrado, este tendrá acceso de por vida a las películas compradas, las cuales podrá visualizar desde cualquier dispositivo, ya sea un móvil, tableta u ordenador debido a que la aplicación tiene un diseño completamente responsivo. También podrá ver sus datos personales, donde podrá modificarlos, ver su historial de compra, descargar sus facturas, entre otras funcionalidades. Todas las películas disponibles a la venta están solamente en formato digital. La aplicación web estará disponible en varios idiomas. Tendrá un apartado de contacto, preguntas frecuentes, política de datos personales, términos y condiciones y sobre nosotros.

Para el desarrollo del TFG se tiene previsto hacer uso de SCRUM como metodología de desarrollo ágil, ya que ayudará mucho en la creación e implementación de las diferentes fases del desarrollo del proyecto. También se hará uso de una gran variedad de herramientas y tecnologías informáticas para poder desarrollar la aplicación.

Como resultados del desarrollo del proyecto, se espera tener una aplicación web con un buen diseño, que sea interactiva y de agrado para los usuarios finales. Que todas las funcionalidades implementadas sean útiles. Que se pueda realizar la compra de las películas sin ninguna dificultad y que se pueda visualizar las mismas desde cualquier navegador web. Que el acceso a la web pueda hacerse a través de un dominio público y que el tráfico esté cifrado mediante un certificado válido para proteger los datos de los usuarios.

1.1. OBJETIVOS INICIALES

Desde el punto de vista de desarrollo:

- Desarrollo de una aplicación web destinada a la venta de películas en formato digital.
- Desarrollo del Front-end, ya sea el diseño y desarrollo de una interfaz gráfica de usuario, teniendo en cuenta conocimientos obtenidos de asignaturas como Diseño de Interfaces de Usuarios (DIU) y aplicando conceptos como la experiencia de usuario (UX).
- Desarrollo del Back-end de la aplicación, donde se manejarán todo lo relacionado a la lógica de la aplicación, ya sea el servidor web, la base de datos, la seguridad, etc.
- Creación de una pasarela de pago.
- Aprendizaje de nuevas tecnologías y nuevos conceptos de desarrollo software.

Desde el punto de vista del usuario:

- Desarrollo de una plataforma web donde los usuarios puedan comprar películas y poder visualizarlas en cualquier momento.
- Poder crear una lista de películas favoritas y guardarla para su posterior compra o acceso a ella para ver el tráiler, descripción, reparto de actores u otra información.
- Poder usar la web de forma fácil sin necesidad de un manual de usuario para ello.
- Poder realizar la compra de las películas a través de una tarjeta de crédito/débito o una cuenta de PayPal.
- Tener acceso a su historial de compra y descargar sus facturas en cualquier momento.
- La posibilidad de modificar sus datos personales.

1.2. COMPETENCIAS ESPECÍFICAS CUBIERTAS

La competencia específica seleccionada fue la CII08 (Capacidad para analizar, diseñar, construir y mantener aplicaciones de forma robusta, segura y eficiente, eligiendo el paradigma y los lenguajes de programación más adecuados)

Esta competencia se ha cubierto en el TFG, ya que se ha tenido que hacer una análisis, diseño e implementación de una aplicación web bastante completa. MoviesHD es robusta por su desarrollo con herramientas como Amazon Web Services (AWS), Google, Spring Boot, entre otras que son importantes y consolidadas en el desarrollo de software a escala mundial.

Segura por las medidas que se toman en la aplicación web como un certificado de capa de conexión segura/seguridad de la capa de transporte (SSL/TLS) para cifrar la conexión entre el cliente y el servidor, seguridad haciendo uso de los roles de usuario con Spring Profile que permite darles acceso a ciertas funcionalidades de la web.

Eficiente, debido al uso de los recursos necesarios en cada momento, como AWS que hace uso de un balanceador para repartir la carga dependiendo de las peticiones que se haga en la web y así evitar que se caiga o ralentice la aplicación.

En cuanto al paradigma y lenguaje de programación se hace uso del framework Spring Boot que proporciona un completo modelo de programación y configuración para aplicaciones empresariales modernas basadas en Java, en cualquier tipo de plataforma de implementación y hace fácil crear aplicaciones que se puede "simplemente ejecutar".

1.3. APORTACIONES

El TFG MoviesHD aporta a nuestro entorno socio-económico una Plataforma Web donde poder comprar películas en formatos digital. En dicha plataforma se ofrece un trato más personal a los usuarios, ya que está especializada en la venta de películas, cosas que la mayoría de las compañías actuales tiene como una de sus funcionalidades, pero no centradas en esa solamente.

En esta plataforma, el usuario puede hacer uso de ella sin tener una cuenta creada, por lo que puede ver el catálogo de películas a la venta y ver los avances de las mismas. El usuario puede también crearse una cuenta, ya sea a través de un correo electrónico y una contraseña, o utilizando proveedores de servicios como Google o Facebook.

Por otro lado, puede crear una lista de películas favoritas para poder visualizar sus detalles en cualquier momento. También puede realizar la compra de las películas con una tarjeta de crédito/débito o mediante una cuenta de PayPal y podrá visualizar en cualquier momento las películas compradas, ya sea desde un dispositivo móvil, tableta u ordenador.

CAPÍTULO 2

ESTADO ACTUAL DEL TEMA

En este capítulo se presenta un estudio de la situación actual de plataformas digitales donde el usuario puede ver películas, series y documentales, ya sea a través de una suscripción o mediante el alquiler o compra de las mismas.

2.1. PLATAFORMAS DIGITALES

Aquí se muestra una comparativa de las principales plataformas digitales en la actualidad, así como sus características relacionadas con los servicios que ofrecen, el catálogo, los dispositivos donde poder visualizar el contenido y los métodos de pago.

2.1.1. Netflix



Figura 2.1: Netflix

Servicios

- Los primeros 30 días son gratis para los nuevos suscriptores.
- Suscripción mensual Básica 7.99 €
- Suscripción mensual Estándar 10.99 €
- Suscripción mensual Premium 13.99 €
- Tarjetas de regalo.
- Poder descargar ciertas películas y series en la aplicación de Netflix para verlas más tarde sin conexión.

Catálogo

- Series, películas, documentales y otros contenidos. galardonados
- Cada mes se añaden más series y películas.
- Originales de Netflix.

Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV, Chromecast.
- Smart TV: Hisense, LG, Panasonic, Philips, Samsung, Sony, Toshiba.
- Videoconsolas: PS3, PS4, Wii U, XBOX 360, XBOX ONE
- Decodificadores: Vodafone
- Reproductores de Blu-Ray
- Smartphones y tabletas: iOS, Android, Windows Phone.
- Aplicación para Windows.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjeta de crédito o débito VISA o MasterCard, American Express.
- PayPal
- Código de tarjeta de regalo.

2.1.2. HBO



Figura 2.2: HBO

Servicios

- El primer mes es de prueba.
- Suscripción mensual 7.99 €

CATÁLOGO

- Series y películas exclusivas de HBO.
- Contenido de otros proveedores como Warner, FOX, SONY y Disney.
- Contenido para los más pequeños de la casa.

• Reproductores multimedia de Streaming: Apple TV, Chromecast.

■ Smart TV: LG, Samsung, Android TV.

■ Videoconsolas: PS3, PS4.

Decodificadores: Vodafone

Smartphones y tabletas: iOS, Android.

MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjeta de crédito o débito VISA o MasterCard, American Express.
- PayPal
- Código promocional.

2.1.3. Amazon Prime Video



Figura 2.3: Amazon Prime Video

Servicios

- Período de 30 días de prueba para miembros nuevos.
- Suscripción mensual con un precio de lanzamiento de 2.99 € mensuales durante los primeros seis meses de la suscripción. La suscripción a Prime Video se renueva automáticamente con un coste de 5.99 € mensuales a partir del séptimo mes de suscripción.
- Si ya tienes una suscripción a Amazon Prime, Prime Video estará disponible sin coste adicional.
- Membresía de Amazon Prime Student, programa de seis meses de prueba que incluye transmisión ilimitada de películas y programas de TV Prime, entre otras muchas cosas. Una vez finalizado el período de prueba de seis meses, la membresía de Prime Student lo hace elegible para recibir el 50 % de descuento en Amazon Prime.
- Opción de descarga de títulos de Prime Video en dispositivos móviles para verlos sin conexión.

Catálogo

- Amplio contenido de películas y series.
- Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV, Amazon Fire TV, Roku TV.
- Smart TV: Sony, Samsung, LG, Philips, Panasonic.
- Videoconsolas: PS3, PS4, XBOX ONE, Nvidia Shield.
- Decodificadores: Vodafone
- Smartphones y tabletas: iOS, Android.
- Tablet Fire de Amazon.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjetas de crédito o débito: VISA, Amazon Store Card, Amazon Credit Builder, MasterCard/EuroCard, Discover Network, American Express, Diner's Club (direcciones de facturación solamente de Estados Unidos), JCB, China UnionPay (tarjeta de crédito solamente).
- Amazon Rewards Visa Signature Card.
- Amazon Business American Express Card.
- Tarjetas de prepago de VISA, MasterCard o American Express.
- Amazon Payments.

2.1.4. Filmin



Figura 2.4: Filmin

Servicios

- Suscripción Mensual Básica 7,99 €
- Suscripción Mensual Plus 14,99 €. Incluye 3 vales que te permiten ver los últimos estrenos marcados con el diamante.
- Suscripción Anual Básica 84 €
- Suscripción Anual Plus 120 €. Incluye 25 vales.
- Alquiler entre 0,95 € y 3,95 €
- Regalar suscripciones.
- Código promocional para ver una película.

 Visualización del contenido sin conexión solamente para series en las aplicaciones para iOS y Android.

Catálogo

- Disponibles más de 10000 títulos.
- Cada día de la semana se producen estrenos en Filmin. Algunos de ellos son títulos clásicos, otros son novedades que han tenido una vida exitosa en cines.
- Se estrenan películas de forma simultánea en cine y en Filmin.
- Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV.
- Smart TV: LG, Samsung, Sony, Philips, Sharp, Android TV.
- Videoconsolas: PS3, PS4.
- Smartphones y tabletas: iOS, Android.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjeta de crédito o débito VISA o MasterCard.
- PayPal

2.1.5. MUBI



Figura 2.5: MUBI

SERVICIOS

- Una semana gratis.
- Suscripción mensual 8.99 €
- Alquiler de películas.
- Regala una nueva película cada día durante un año por 71 €
- Descarga temporal de las películas a través de la aplicación en iOS y Android para verlas sin conexión.

CATÁLOGO

- Una película nueva cada día.
- Siempre hay 30 películas elegidas cuidadosamente a mano para ver o descargar. Desde joyas olvidadas al cine recién estrenado. Desde clásicos de culto a obras maestras galardonadas.
- Estrenos de MUBI.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV, Chromecast, Amazon Fire TV, Roku TV.
- Smart TV: Sony, Samsung, LG, Android TV.
- Videoconsolas: PS3, PS4, Nvidia Shield.
- Smartphones y tabletas: iOS, Android.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjetas de crédito o débito: American Express, MasterCard, Maestro y VISA (excepto tarjeta débito VISA Electron).
- PayPal

2.1.6. FlixOlé



Figura 2.6: FlixOlé

Servicios

- Catorce días de prueba a todos los usuarios nuevos , donde tendrán acceso total a todo el contenido durante ese período de prueba.
- Suscripción mensual 2.99 €
- Descuento por código promocional.

CATÁLOGO

- Primera plataforma española con más de 3000 películas y series.
- Cine español, europeo y americano, clásico y reciente, de todos los géneros.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV, Chromecast.
- Smart TV: Samsung, Android TV.
- Smartphones y tabletas: iOS, Android.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

Tarjeta de crédito o débito VISA o MasterCard, American Express.

2.1.7. Sky



Figura 2.7: Sky

Servicios

- Prueba gratuita de 1 mes para nuevos clientes.
- Suscripción mensual 6.99 €
- Descargas disponibles para una selección limitada de series y películas, solo mediante acceso vía Wi-Fi en el territorio de la Unión Europea. Solo en dispositivos móviles y tabletas con sistema operativo Android y iOS.

CATÁLOGO

- Series, películas y documentales.
- Canales de pago: FOX, LaLiga 1|2|3 TV, Disney Junior, entre otros.
- Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV, Sky TV Box, Agile TV Box de Yoigo, Chromecast.
- Smart TV: Samsung, LG.
- Smartphones y tabletas: iOS, Android.
- Videoconsolas: PS3, PS4, XBOX ONE.

MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjeta de crédito o débito. El único requisito es que la tarjeta de débito o crédito elegida sea emitida por una entidad bancaria española.
- PayPal

2.2. ALTERNATIVAS RELACIONADAS CON MOVIESHD

2.2.1. Rakuten TV



Figura 2.8: Rakuten TV

Servicios

- Contenido disponible en alquiler (tarifas entre 0,99 € y 4,99 €) y/o compra.
- Suscripción a Rakuten Wuak: primer mes gratuito, resto de cuotas 6'99 €/mes, sin compromiso de permanencia. Las películas y series incluidas en la suscripción son las que aparecen en el apartado "Rakuten Wuaki"
- Rakuten Super Puntos: recompensas por usar los servicios de Rakuten. Son una moneda virtual, lo que significa que puede utilizarlos para comprar productos en Rakuten TV.
- Visualización de cierto contenido sin conexión a través de la aplicación de Rakuten TV en la mayoría de los dispositivos Android e iOS.

CATÁLOGO

- Las mejores películas y series para el usuario, desde producciones nacionales hasta grandes éxitos de Hollywood.
- Los mejores estrenos y grandes clásicos.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Chromecast.
- Smart TV: Hisense, LG, Panasonic, Philips, Samsung, Sony, Hisense.
- Smartphones y tabletas: iOS, Android.
- Videoconsolas: PS3, PS4, XBOX 360, XBOX ONE.

MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjeta de crédito o débito VISA o MASTERCARD, de entidades bancarias que operen en España y Andorra, exceptuando las tarjetas MAESTRO.
- PayPal
- Rakuten Super Puntos.

2.2.2. Google Play Movies



Figura 2.9: Google Play Movies

Servicios

- Alquiler de películas.
- Compra de películas (disponible en 25 países) y series (solo están disponibles en Estados Unidos, Japón, Australia y el Reino Unido).
- Tarjetas de regalo.
- Posibilidad de descargar el contenido para poder verlo posteriormente sin necesidad de una conexión a Internet.

Catálogo

- Unos de los catálogos de películas y series más amplio a nivel mundial.
- Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Chromecast.
- Smartphones y tabletas: iOS, Android.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjeta de crédito o débito VISA o MasterCard, American Express.
- PayPal
- Paysafecard
- Tarjetas de regalo.

2.2.3. **Amazon**



Figura 2.10: Amazon

Servicios

- Alquiler o compra de películas, series, TV Shows.
- Visualización de contenido con Amazon Prime.

CATÁLOGO

- Amplio contenido de películas, series, TV Shows.
- Originales de Amazon.
- Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Reproductores multimedia de Streaming: Apple TV, Amazon Fire TV, Roku TV.
- Smartphones y tabletas: iOS, Android.
- Tablet Fire de Amazon.
- Smart TV: Sony, Samsung, LG, Philips, Panasonic.
- Videoconsolas: PS3, PS4, XBOX ONE, Nvidia Shield.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjetas de crédito o débito: VISA, Amazon Store Card, Amazon Credit Builder, MasterCard/EuroCard, Discover Network, American Express, Diner's Club (direcciones de facturación solamente de Estados Unidos), JCB, China UnionPay (tarjeta de crédito solamente).
- Amazon Rewards Visa Signature Card.
- Amazon Business American Express Card.
- Tarjetas de prepago de VISA, MasterCard o American Express.
- Amazon Payments.



Figura 2.11: Microsoft Films & TV

2.2.4. Microsoft Films & TV

Servicios

- Alquiler o compra de películas.
- Compra de series y programas de televisión.
- Tarjetas de regalo.

Catálogo

- Amplia colección de películas, series y programas de televisión.
- Contenido para los más pequeños de la casa.

Dispositivos donde visualizar el contenido

- Smartphones y tabletas: Windows Phone.
- HoloLens.
- Videoconsolas: XBOX 360, XBOX ONE.
- PC con Windows 8, Windows 8.1, Windows 10 o una versión posterior.
- MAC, PC: Navegadores web.

MÉTODOS DE PAGO

- Tarjetas de crédito o débito.
- Adeudo directo
- PayPal
- Teléfono móvil
- Tarjetas de regalo.

16 2.3. EVALUACIÓN

2.3. EVALUACIÓN

A partir de las comparativas realizadas en los apartados anteriores respecto a la situación actual del mercado en cuanto a la distribución de contenido audiovisuales, ya sea películas, series, programas de televisión o documentales, se ha llegado a la conclusión de que la gran mayoría ofrecen los mismos servicios, métodos de pago y dispositivo para la visualización del contenido. En cuanto al catálogo ofertado y el poder visualizarlo sin conexión a Internet es donde hay más diferencias.

Por tanto, para que dichas plataformas se diferencien entre ellas, depende principalmente de que los precios sean competitivos. EL catálogo debe ser muy variable y que se pueda visualizarse sin acceso a Internet. El contenido debe tener una alta disponibilidad y buena calidad en streaming y que la plataforma tenga una buena interfaz gráfica y experiencia de usuario.

Por otro lado, MoviesHD actualmente ofrece servicios, catálogo y métodos de pagos similares al mercado actual. En cuanto a los dispositivos donde poder visualizar el contenido es donde tiene menos opciones, los cuales son una de las extensiones del proyecto.

PLANIFICACION DEL PORTAL WEB

3.1. PLANIFICACIÓN

Para la elaboración del proyecto se siguió un proceso de desarrollo guiado por el Modelo Iterativo e Incremental (Figura 3.1), el cual se planifica en diversos **bloques temporales** (en el caso de SCRUM de un mes natural o hasta de dos semanas, si así se necesita) llamados iteraciones.

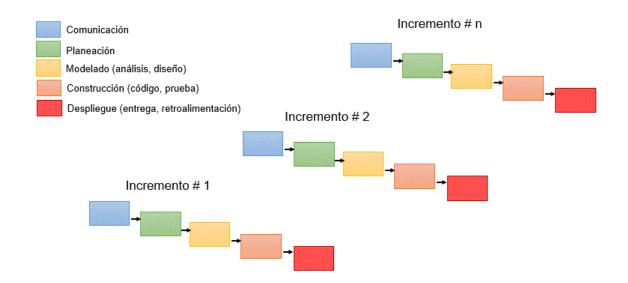


Figura 3.1: Modelo Iterativo e Incremental

Las iteraciones se pueden entender como miniproyectos: en todas las iteraciones se repite un proceso de trabajo similar (de ahí el nombre "iterativo") para proporcionar un resultado completo sobre producto final, de manera que el cliente pueda obtener los beneficios del proyecto de forma incremental. Para ello, cada requisito se debe completar en una única iteración: el equipo debe realizar todas las tareas necesarias para completarlo (incluyendo pruebas y documentación) y que esté preparado para ser entregado al cliente con el mínimo esfuerzo necesario. De esta manera no se deja para el final del proyecto ninguna actividad arriesgada relacionada con la entrega de requisitos.

En cada iteración el equipo **evoluciona el producto** (hace una entrega incremental) a partir de los resultados completados en las iteraciones anteriores, añadiendo nuevos objetivos/requisitos o mejorando los que ya fueron completados. Un aspecto fundamental para guiar el desarrollo iterativo e incremental es la **priorización de los objetivos/requisitos en función del valor que aportan al cliente**.

A continuación, se desglosa de manera detallada el trabajo del TFG en fases, con su duración

18 3.1. PLANIFICACIÓN

estimada en horas (total 300). Cada fase a su vez, se encuentra desglosada en tareas concretas con su nombre y descripción.

Fases	Est.	Tareas
		(nombre, descripción)
Creación inicial del Product Backlog	15	Tarea 1.1: Análisis de la situación actual (5 hrs)
		Tarea 1.2: Primera toma de requisitos de usuario (historias de usuarios) y priorización de necesidades. (10 hrs)
2. Sprint 0	85	Tarea 2.1: Aprendizaje y familiarización con las nuevas herramientas y tecnologías: Spring Boot, Thymeleaf, Spring Security, API PayPal, Braintree, Google y Facebook Sign-in API, AWS, etc. (60 hrs)
		Tarea 2.2: Diseño de la base de datos (10 hrs)
		Tarea 2.3: Integración de los productos de AWS que usa- rá la aplicación para desplegarla en los servidores, uso del protocolo https para cifrar la comunicación entre la aplicación web y los servidores de Amazon, conexión con la base de datos de Amazon RDS (15 hrs)
3. Sprint 1	43	Tarea 3.1: Creación de la página de inicio con su diseño y sus diferentes secciones. (16 hrs)
		Tarea 3.2: Creación del formulario de registro e inicio de sesión tradicional, al igual que el botón de cerrar sesión. (13 hrs)
		Tarea 3.3: Creación del buscador de películas y el filtro de búsqueda por género, idioma, etc. (10 hrs)
		Tarea 3.4: Creación de la página de detalle de las películas, con su ventana modal para reproducir el trailer. (4 hrs)
4. Sprint 2	62	Tarea 4.1: Creación del carro de la compra con su vista de detalle. (8 hrs)
		Tarea 4.2: Creación de la pasarela de pago, el envío de la confirmación de la compra y la página del historial de compras. (32 hrs)
		Tarea 4.3: Creación de la página para acceder a las películas compradas, al igual que el reproductor de vídeo. (14 hrs)
		Tarea 4.4: Implementación de la lógica y de las vistas para el proceso de recuperación de la contraseña. (8 hrs)

Fases	Est.	Tareas (nombre, descripción)
5. Sprint 3	55	Tarea 5.1: Implementación del sistema de inicio de sesión con Google y Facebook. (10 hrs)
		Tarea 5.2: Creación la sección de listado de películas favoritas con su página de detalles. (11 hrs)
		Tarea 5.3: Implementación la funcionalidad de editar los datos personales. (4 hrs)
		Tarea 5.4: Creación de las páginas de preguntas frecuentes, políticas de protección de datos, términos y condiciones, sobre nosotros, etc. (10 hrs)
		Tarea 5.5: Implementación de la paginación en las diferentes secciones de la web. (8 hrs)
		Tarea 5.6 Disponibilidad de la web en varios idiomas. (12 hrs)
Documentación / Presentación	40	Redacción de la memoria y la presentación.

Tabla 3.1: Tabla que refleja las distintas fases del desarrollo del proyecto

3.2. METODOLOGÍA

Como metodología de desarrollo ágil se hace uso de SCRUM para una mejor gestión de todas las fases del proyecto. SCRUM es un proceso en el que se aplican de manera regular un conjunto de buenas prácticas para trabajar colaborativamente, en equipo, y obtener el mejor resultado posible de un proyecto.

En SCRUM se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, SCRUM está especialmente indicado para proyectos en entornos complejos, donde se necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

El proceso

El desarrollo se realiza de forma iterativa e incremental. Cada iteración, denominada Sprint, tiene una duración preestablecida de entre 2 y 4 semanas, obteniendo como resultado una versión del software con nuevas prestaciones listas para ser usadas. En cada nuevo Sprint, se va ajustando la funcionalidad ya construida y se añaden nuevas prestaciones priorizándose siempre aquellas que aporten mayor valor de negocio.

■ **Product Backlog:** Conjunto de requisitos denominados historias descritos en un lenguaje no técnico y priorizados por valor de negocio, o lo que es lo mismo, por retorno de inversión considerando su beneficio y coste. Los requisitos y prioridades se revisan y ajustan durante el curso del proyecto a intervalos regulares.

20 3.2. METODOLOGÍA

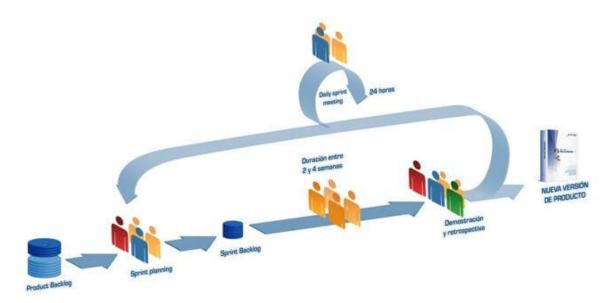


Figura 3.2: Proceso de SCRUM

- Sprint Planning: Reunión durante la cual el Product Owner presenta las historias del backlog por orden de prioridad. El equipo determina la cantidad de historias que puede comprometerse a completar en ese Sprint, para en una segunda parte de la reunión, decidir y organizar cómo lo va a conseguir.
- Sprint: Iteración de duración prefijada durante la cual el equipo trabaja para convertir las historias del Product Backlog a las que se ha comprometido, en una nueva versión del software totalmente operativo.
- Sprint Backlog: Lista de las tareas necesarias para llevar a cabo las historias del Sprint.
- Daily Sprint Meeting: Reunión diaria de cómo máximo 15 min. en la que el equipo se sincroniza para trabajar de forma coordinada. Cada miembro comenta que hizo el día anterior, que hará hoy y si hay impedimentos.
- **Demo** y retrospectiva: Reunión que se celebra al final del Sprint y en la que el equipo presenta las historias conseguidas mediante una demonstración del producto. Posteriormente, en la retrospectiva, el equipo analiza qué se hizo bien, qué procesos serían mejorables y discute acerca de cómo perfeccionarlos.

Roles

En SCRUM, el equipo se focaliza en construir software de calidad. La gestión de un proyecto SCRUM se centra en definir cuáles son las características que debe tener el producto a construir (qué construir, qué no y en qué orden) y en vencer cualquier obstáculo que pudiera entorpecer la tarea del equipo de desarrollo.

El equipo SCRUM está formado por los siguientes roles:

 SCRUM Master: Persona que lidera al equipo guiándolo para que cumpla las reglas y procesos de la metodología. Gestiona la reducción de impedimentos del proyecto y trabaja con el Product Owner para maximizar el Return On Investment (ROI).

- **Product Owner (PO):** Representante de los accionistas y clientes que usan el software. Se focaliza en la parte de negocio y él es responsable del ROI del proyecto (entregar un valor superior al dinero invertido). Traslada la visión del proyecto al equipo, formaliza las prestaciones en historias a incorporar en el Product Backlog y las vuelve a priorizar de forma regular.
- **Development Team:** Grupo de profesionales con los conocimientos técnicos necesarios y que desarrollan el proyecto de manera conjunta llevando a cabo las historias a las que se comprometen al inicio de cada sprint.

Debido a que es un trabajo unipersonal, solo se hará uso como proceso del desarrollo de SCRUM: el Product Backlog, Sprint, Sprint Backlog y la Demo y retrospectiva. En cuanto al Role desempañado será el de Development.

Creación de las historias de usuario

Las historias de usuario se usan para capturar la esencia del valor de negocio de un sistema usando lenguaje cotidiano. Una historia de usuario describe una funcionalidad, están centradas en necesidades y responde a preguntas como ¿qué quiere el usuario?, ¿por qué lo quiere?

A continuación, se muestra el Product Backlog final con todas las historias de usuarios para la creación de la Aplicación Web MoviesHD. La prioridad tiene un rango de 1-100 siendo el 100 la historia de usuario más prioritaria y la estimación es en horas.

Nº	Nombre	Pri.	Est.	Descripción
1	Página de inicio	100	16	Como usuario quiero que al acceder a la página de inicio de la aplicación, esta me muestre las películas más compradas, más vistas y más favoritas en diferentes secciones de la web.
2	Sistema de inicio de sesión tradicional	90	8	Como usuario quiero poder acceder a mi cuenta a través de un usuario y contraseña.
3	Sistema de inicio de sesión con Google o Facebook	55	10	Como usuario quiero poder acceder a mi cuenta a través de Google o Facebook.
4	Sistema de registro	85	4	Como usuario quiero poder registrarme en la web.
5	Sistema de cierre de sesión	90	1	Como usuario quiero poder cerrar la sesión de mi cuenta.
6	Buscador de películas	95	8	Como usuario quiero poder buscar películas por su nombre.
7	Filtro de búsqueda de películas	95	4	Como usuario quiero poder filtrar la búsqueda de películas por género e idioma.
8	Paginación	35	8	Como usuario quiero que la web me muestre las películas con una paginación para no mostrar todo en la misma página.

22 3.2. METODOLOGÍA

Nº	Nombre	Pri.	Est.	Descripción
9	Detalle de la pelícu- la	95	4	Como usuario quiero poder ver la descripción, duración, calidad digital, reparto y avance de las películas.
10	Lista de favoritos	50	7	Como usuario quiero poder añadir películas a mi lista de favoritos para poder ver el avance o com- prarlas más adelante.
11	Detalle lista de favo- ritos	45	4	Como usuario quiero poder ver el detalle de mi lista de favoritos.
12	Carro de la compra	80	8	Como usuario quiero poder añadir películas a mi carro de la compra.
13	Detalle carro de la compra	80	4	Como usuario quiero poder ver el detalle de mi carro de la compra y que me permita modificarlo.
14	Pasarela de pago	75	20	Como usuario quiero poder realizar el pago de mi carro de la compra, ya sea con una cuenta de PayPal o con una tarjeta de crédito o débito.
15	Confirmación de compra	70	6	Como usuario quiero recibir una confirmación por correo electrónico del detalle de mis compras.
16	Editar datos perso- nales	40	5	Como usuario quiero poder editar algunos de mis datos personales.
17	Recuperar contrase- ña	60	8	Como usuario quiero tener la opción de recuperar mi contraseña en caso de olvido o pérdida.
18	Historial de com- pras	60	6	Como usuario quiero poder ver mi historial de compra para descargar las facturas de las mismas.
19	Acceso películas compradas	65	8	Como usuario quiero poder acceder a mis películas y poder visualizarlas en cualquier momento.
20	Multi-idioma	25	12	Como usuario quiero poder cambiar el idioma de la web, ya sea en español, inglés, francés o alemán.
21	Preguntas frecuentes	30	2	Como usuario quiero poder ver un listado de preguntas frecuentes.
22	Contacto	30	2	Como usuario quiero poder contactar con MoviesHD para cualquier duda que tenga.
23	Política de datos personales	30	2	Como usuario quiero saber la política de protección de mis datos personales.
24	Sobre nosotros	30	1	Como usuario quiero poder saber un poco más sobre MoviesHD.
25	Términos y condi- ciones	30	2	Como usuario quiero saber los términos y condiciones para el uso de MoviesHD.

Tabla 3.2: Tabla que refleja las historias de usuarios del Product Backlog

3.3. PRESUPUESTO

A continuación se expondrán los costes necesario para la elaboración de este proyecto, el cual está dividido en las siguientes cuatro categorías:

- Costes de personal: Sueldos del personal involucrado en el desarrollo del proyecto.
- Costes de hardware: Equipos necesarios para la realización del proyecto: Ordenadores, monitores, etc.
- Costes de software: Herramientas y tecnologías utilizadas en el desarrollo del proyecto.
- Otros costes: Todo lo relacionado con el alquiler del local de trabajo, material de oficina, impuestos de electricidad, agua, etc.

3.3.1. Costes de Personal

A continuación se muestran los costes relacionados a los sueldos mensuales del personal que debería estar involucrado en el desarrollo del proyecto.

Personal	Sueldo (€)	
Jefe de proyecto	3115	
Diseñador de interfaces de usuario	1732	
Desarrollador Front-end	2291	
Desarrollador Back-end	2867	
Total	10005	

Tabla 3.3: Tabla que refleja los costes mensuales del personal

3.3.2. Costes de Hardware

En los costes de hardware vendrán incluidos todos los equipos necesario para el desarrollo del proyecto teniendo en cuenta el número de personal involucrado.

Equipo	Unidades	Precio unitario (€)	Total (€)
Ordenador de sobremesa	4	1006	4024
Monitor	4	175	700
Conjunto de teclado y ratón	4	30	120
Impresora laser multifuncional	1	186	186
Coste total			5030

Tabla 3.4: Tabla que refleja los costes hardware del proyecto

3.3.3. Costes de Software

Los productos software que se han utilizado en el desarrollo del proyecto no han tenido ningún tipo costes, debido a que los mismos son de código abierto (Open Source) o bajo licencias del MIT.

Por otra parte, los servicios de AWS utilizados tampoco han tenido coste alguno debido al período de un año gratuito en el uso de los mismos.

24 3.3. PRESUPUESTO

3.3.4. Otros costes

En este apartado se incluyen otros gastos mensuales necesarios como el alquiler de una oficina o el de suministros, que no entran en las categorías de costes descritas anteriormente.

Descripción	Coste (€)
Alquiler de oficina	1200
Material de oficina	200
Internet	60
Agua	30
Electricidad	100
Total	1590

Tabla 3.5: Tabla que refleja otros costes mensuales del proyecto

3.3.5. Costes totales

En la siguiente tabla se mostrará el total de todos los costes durante el período de un año para financiar la elaboración del Portal Web MoviesHD.

Costes	1 año (€)
Personal	120060
Hardware	5030
Software	0
Otros	19080
Total	144170

Tabla 3.6: Tabla que refleja los costes totales del proyecto durante el período de un año

DESARROLLO DEL PORTAL WEB

Este capítulo está estructurado por los diferentes Sprint vistos en el capítulo de planificación del portal web. En el Sprint 0 es donde se aprende y se familiariza con las herramientas y tecnologías a usar en el proyecto, también es donde se diseña la base de datos de la aplicación web y se configuran los productos de AWS para su uso en la web.

Por otro lado, están los Sprint 1, 2 y 3, en los cuales, al principio de cada uno de ellos se muestran las historias de usuarios que lo componen. Cada historia de usuario contiene una descripción de la misma, al igual que imágenes que la representan y un listado de tareas por el que está compuesto su desarrollo.

Al final de cada Sprint se encuentran los apartados de retrospectiva y mejoras del mismo. En la retrospectiva se contemplan las historias de usuarios a destacar en el Sprint y las dificultades en la implementación de algunas de las historias de usuarios que lo componen. En el segundo apartado se muestra un listado de las mejoras realizadas al Sprint.

4.1. SPRINT 0

Este sprint está dividido en tres fases que se pasará a detallar más adelante, las cuales son: aprendizaje y familiarización con las nuevas herramientas y tecnologías a usar en el proyecto, el diseño de la base de datos y la integración con los productos de Amazon Web Services (AWS).

4.1.1. Aprendizaje y familiarización con las nuevas herramientas y tecnologías

Esta tarea es una de las más importantes del desarrollo de la aplicación web, ya que es la base para la creación de MoviesHD. Aquí es donde se aprende y se familiariza con las distintas herramientas y tecnologías a usar en la creación de la web. A continuación, se detalla lo aprendido y usado en el desarrollo del proyecto.

HERRAMIENTAS Y TECNOLOGÍAS:

- Integrated Development Environment (IDE) IntelliJ IDEA[1].
- **Sourcetree**[2], para el uso del sistema de control de versiones GIT[3] y el esquema de trabajo GitFlow.
- MySQL Workbench[4] como gestor de base de datos.
- AWS[5] donde se hará uso de:
 - Amazon Simple Storage Service (Amazon S3)[6], que es un servicio de almacenamiento de objetos en la nube.

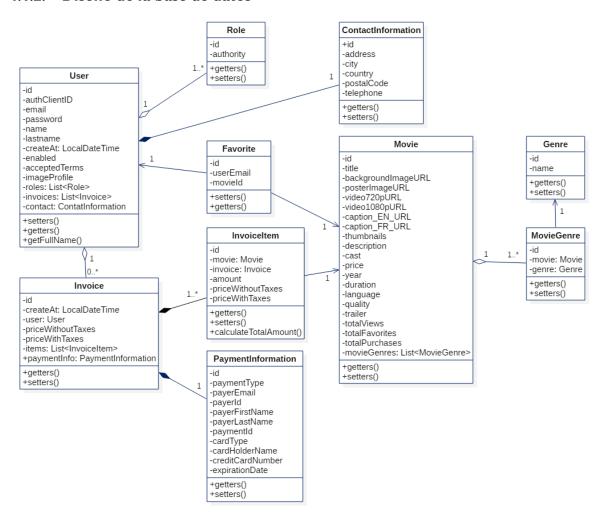
26 4.1. SPRINT 0

 Content Delivery Network (CDN) Amazon CloudFront[7], que es un servicio rápido de red de entrega de contenido.

- Amazon Relational Database Service (Amazon RDS)[8] para configurar y escalar bases de datos relacionales en la nube.
- Amazon Elastic Beanstalk[9], para implementar y escalar servicios y aplicaciones web desarrolladas en Java, .NET, PHP, Node.js, Python, Ruby, etc.
- Amazon Elastic Compute Cloud (Amazon EC2)[10], que proporciona capacidad informática en la nube de forma segura y de tamaño modificable.
- Domain Name System (DNS) **Amazon Route 53[11**], que es un servicio de sistema de nombre de dominios web escalable y de alta disponibilidad en la nube.
- Amazon Certificate Manager[12], que permite aprovisionar, administrar e implementar con facilidad certificados de capa de conexión segura de la capa de trasporte (SSL/TLS).
- Identity and Access Management (IAM)[13], que permite administrar el acceso a los servicios y recursos de Amazon de manera segura. Con IAM, puede crear y administrar usuarios y grupos de AWS.
- Amazon Virtual Private Cloud (Amazon VPC)[14], permite aprovisionar una sección de la nube de AWS aislada de forma lógica.
- Java y JavaScript como los lenguajes de programación principales.
- Tecnologías de Spring:
 - Spring Boot[15], un framework de Java con el cual se desarrollará la aplicación web.
 - Spring Data[16], el cual facilita el uso de tecnologías de acceso a datos, base de datos relacionales y no relacionales, dentro del cual se hará uso de Spring Data JPA para la implementación de repositorios basados en Java Persistence API (JPA).
 - **Spring Security**[17], un framework potente y altamente personalizable de autenticación y control de acceso.
 - **Spring Mobile**[18], un framework que proporciona capacidades para detectar el tipo de dispositivo que realiza una petición en el sitio web.
- Thymeleaf[19], moderno motor de plantillas Java para servidores, tanto para entornos web como autónomos.
- Maven[20], herramienta de software para la gestión y construcción de proyectos en Java.
- Bootstrap[21], librería de componentes front-end en HTML, CSS y JS para el desarrollo de proyectos web.
- **jQuery**[22], librería de javascript rápida, pequeña y con muchas funciones que hace que cosas como la manipulación del Document Object Model (DOM), el manejo de eventos, animaciones y Asynchronous JavaScript And XML (AJAX) sean mucho más simples.
- jQuery UI[23], conjunto de interacciones de interfaz de usuario, efectos, widgets y temas creados sobre la librería de jQuery.
- Braintree, servicio de PayPal que facilita la aceptación de pagos en una aplicación o sitio web.
- Firebase Authentication[24], sistema de autenticación seguro, compatible con cuentas de correo electrónico y contraseñas, autenticación telefónica, acceso a través de Google, Facebook, Twitter, GitHub y muchos más.

- JSON Web Token (JWT)[25], estándar abierto (RFC 7519) que define una forma compacta y autónoma para transmitir de forma segura información entre las partes como un JavaScript Object Notation (JSON). Esta información puede ser verificada y confiable porque está firmada digitalmente.
- **Swiper**[26], es un control deslizante táctil móvil con transiciones aceleradas por hardware y un sorprendente comportamiento nativo.
- **Plyr**[27], reproductor multimedia simple, accesible y personalizable para Video, Audio, YouTube y Vimeo.
- StarUML[28], sofisticado modelador de software para modelado ágil y conciso.

4.1.2. Diseño de la base de datos



4.1.3. Integración inicial con los servicios de AWS

Para determinados servicios, AWS usa diferentes zonas, por lo que la aplicación MoviesHD hará uso de la zona EU (London), la cual tiene el código (eu-west-2c) para su posterior identificación en futuras configuraciones. Todas las configuraciones se hacen a través de la Consola de Administración de AWS.

Pasos para la integración con los servicios de AWS que usará la aplicación web:

1. Creación de una cuenta de AWS.

28 4.2. SPRINT 1

2. Creación de un usuario IAM. Recomendación por parte de Amazon por motivos de seguridad.

- 3. Creación de un grupo para asignar permisos al usuario creado en el paso anterior.
- 4. Asignar al grupo creado en el paso tres la política administrativa denominada AdministratorAccess, permitiendo a los usuarios del grupo tener acceso de administrador. De aquí en adelante las configuraciones en los servicios de AWS serán configuradas con el usuario IAM creado.
- Creación de una instancia de Amazon RDS para la Base de Datos MoviesHD. El motor es el de MySQL.
- 6. Creación en Amazon S3 de dos buckets, el primero para almacenar las imágenes de las películas y los archivos de subtítulos, el segundo bucket es para almacenar los archivos de vídeos de cada película. Amazon S3 almacena datos a modo de objetos dentro de buckets. Un objeto se compone de un archivo y, de forma opcional, de cualquier metadato que describa dicho archivo.
- 7. Asignación de permisos a los buckets creados en el paso seis. Para cada bucket, se podrá controlar el acceso (quién puede crear, eliminar y enumerar objetos del bucket), elegir la región geográfica en la que Amazon S3 almacenará el bucket y su contenido.
- 8. Creación de dos distribuciones de CloudFront para acceder a los objetos almacenados en los buckets creados en el paso seis.
- 9. Creación de un entorno web en Java dentro de Elastic Beanstalk para desplegar MoviesHD, este entorno hace uso de Amazon EC2 que es donde se desplegará el archivo empaquetado con extensión .jar de la aplicación web.

4.2. **SPRINT 1**



Figura 4.1: Historias de usuario del Sprint 1

A continuación, se pasa a detallar el desarrollo de cada historia de usuario (HU).

4.2.1. HU 1. Página de inicio

Como primera historia de usuario, es aquí donde se crea la estructura base del proyecto, al igual que el head, header y footer para su reutilización en todas las páginas webs de MoviesHD.

Por otro lado, la página de inicio se estructura en diferentes secciones, las cuales son:

- Carrusel de imágenes de las cuatro películas más vistas (Figura 4.2).
- La sección de las diez películas más vistas (Figura 4.3).
- La sección de las diez películas más compradas (Figura 4.3).
- La sección de las diez películas más favoritas (Figura 4.3).

Toda la información e imágenes de cada película mostrada en MoviesHD se obtienen de la web **The Movie DB** (TMDB)[29], la cual es una base de datos de TV y películas creada por la comunidad que se remonta a 2008.

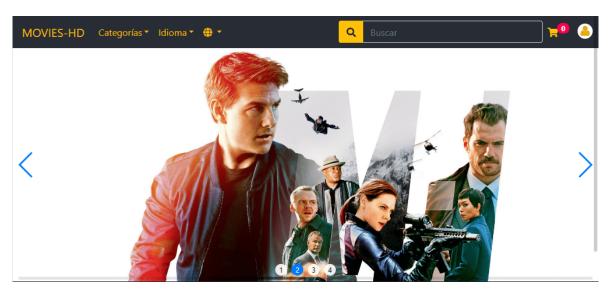


Figura 4.2: Cabecera y carrusel de la página de inicio

- Creación de la estructura base del proyecto en Spring Boot.
- Creación de la disposición de la vista con Thymeleaf, la cual está dividida en el head, header y footer de la web.
- Añadidas las librerías de Bootstrap y jQuery al proyecto.
- Añadida la librería de Fontawesome 5[30] para los iconos de la web.
- Añadida la librería Swiper para el carrusel de imágenes de las cuatro películas más vistas al principio de la página de inicio y las diferentes secciones de la misma.
- Creación de las diferentes opciones en el header de la web, como el nombre de la aplicación, el filtro de búsqueda de las películas por el género e idioma, las opciones de cambiar el lenguaje de la web, el icono del carro de la compra y el menú de opciones.
- Creación de las siguientes hojas de estilo:
 - b4_sidebar.css para cambiar la configuración por defecto del header de Bootstrap en dispositivos con pantalla pequeña.

30 4.2. SPRINT 1

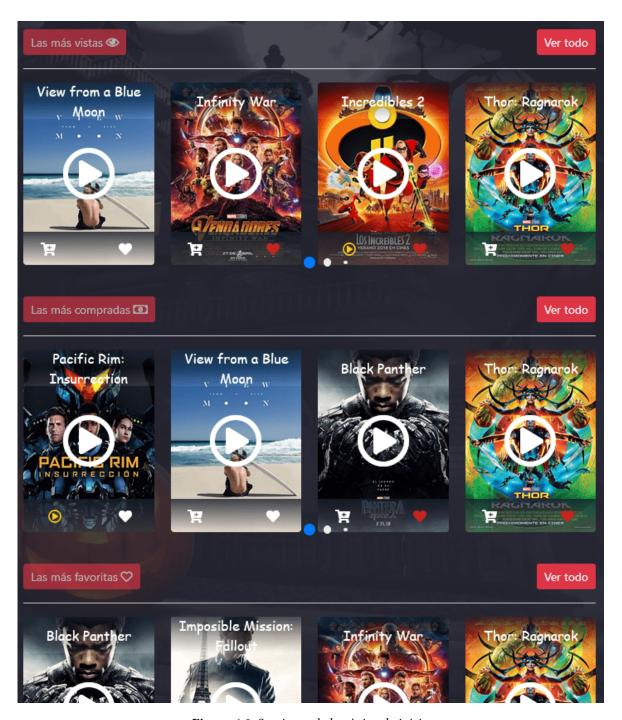


Figura 4.3: Secciones de la página de inicio

- movie.css, para el diseño de las imágenes de las películas, donde aparte de la propia imagen, se muestra el nombre de la película, los iconos del carro de la compra, el de favorito y el de mostrar el detalle de la película.
- styles.css, en el cual irá el resto de las reglas de estilos de la web.
- Creación de los siguientes archivos javascript:
 - *b4_sidebar.js* para cambiar la configuración por defecto del header de Bootstrap en dispositivos con pantalla pequeña.
 - swiper.js para la creación de las diferentes secciones de la página de inicio.
- Añadidas las imágenes de las banderas de los diferentes idiomas que tendrá la web.

- Añadidas las siguientes dependencias al archivo pom.xml:
 - Dicho archivo se trata principalmente del manejo y control de dependencias del proyecto, ya que se hace uso de Maven.
 - *spring-boot-starter-data-jpa* para el Object Relational Mapping (ORM), que consiste en mapear una clase en Java a una tabla en una base de datos y haciendo uso de JPA.
 - mysql-connector-java para trabajar con la base de datos MySQL.
- Añadidas en el archivo application.properties las variables de configuración para la conexión con la base de datos. En este archivo es donde van todas las configuraciones del proyecto.
- Creación de los siguientes elementos en el servidor:
 - Módulo **entities**, donde se almacenarán todos los objetos entidades de la aplicación web que corresponda con una tabla en la base de datos.
 - Entidad Movie, objeto para mapear la tabla donde se almacenará todos los datos de cada película.
 - Módulo Data Access Object (DAO), que como su nombre indica, será para el acceso a los datos de los objetos, en este caso de las entidades.
 - Interfaz IMovieDao para el acceso a la entidad Movie, la cual proporcionará la estructura de los métodos Create, Read, Update and Delete (CRUD), aparte de los creados en un futuro si la aplicación web lo requiere.
 - Módulo services para estructurar los diferentes servicios de las entidades de la aplicación.
 - Interfaz IMovieService, la cual contendrá la estructura de todos los métodos para la
 entidad Movie, ya sea el CRUD, el buscar una película por su nombre, por su idioma,
 por género, las diez más compradas, vistas o favoritas, que son las que se usarán en las
 diferentes secciones de la página de inicio, al igual que las cuatro películas más favoritas
 para el carrusel de imágenes.
 - Servicio MovieServiceImpl que implementará la interfaz mencionada en el punto anterior con todos sus métodos.
 - Módulo controllers para estructurar los diferentes controladores de las entidades para gestionar las peticiones web.
 - Controlador MovieController para gestionar todas las peticiones relacionadas con las películas. Para este Sprint las peticiones incluidas en el controlador son las de obtener las cuatro películas más vistas, las diez más compradas, las diez más favoritas y las diez más vistas, ya que son las diferentes secciones de la página de inicio.
- Añadido al archivo index.html las diferentes secciones de la página de inicio.
- Creación del archivo import.sql para popular la tabla movies con algunas películas. En este archivo es donde irán todas las instrucciones SQL para popular las diferentes tablas de la aplicación que se vayan necesitando durante el desarrollo del proyecto.
- Creación de las siguientes URLs dentro de MovieController:
 - *base-url/movies/favorites/all*, se obtiene de la base de datos y se muestra en la web todas las películas favoritas en orden descendente.
 - base-url/movies/purchased/all, se obtiene de la base de datos y se muestra en la web todas las películas compradas en orden descendente.

32 4.2. SPRINT 1

 base-url/movies/watched/all, se obtiene de la base de datos y se muestra en la web todas las películas vistas en orden descendente.

base-url, es la **Uniform Resource Locator (URL)** base de la página web, que puede ser localhost cuando se está ejecutando en el entorno de desarrollo o www.movies-hd.cf cuando está en producción, debido a que es el dominio público de MoviesHD.

4.2.2. HU 9. Detalle de la película

En esta historia de usuario se muestra una página con el detalle de la película que haya seleccionado el usuario, donde aparte de poder ver el avance de la misma, podrá también añadir la película al carro de la compra o a su lista de películas favoritas si ha iniciado sesión previamente.

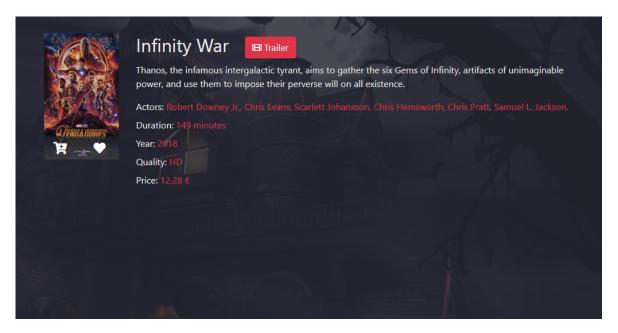


Figura 4.4: Detalle de la película

- Creación de la página movie_detail.html para mostrar el detalle de una película, la cual contendrá diferentes atributos de la entidad Movie, como el nombre, la descripción, el reparto de autores, la duración, el precio y la calidad digital. También se podrá ver el avance de la película, el cual se mostrará en un modal.
- Añadida la url base-url/movie/{id} en MovieController para mostrar el detalle de una película.
 El "id" es el identificador de la película en la base de datos y que se encuentre entre llaves significa que es un valor variable, ya que cada película tiene su propio identificador.
- Añadido el enlace en el icono de reproducir la película para ver el detalle de la misma.
- Añadido en el archivo *styles.css* las reglas de estilos del diseño del modal que muestra el avance de la película y a la información de la misma.
- Creado el archivo *modal.js* para que cuando el modal que muestra el avance de la película se cierre, se detenga la reproducción del vídeo.

4.2.3. HU 6. Buscador de películas

El buscador de películas se ha creado de modo que cuando la web sea visitada desde un dispositivo móvil, dicho buscador se muestre en toda la pantalla (Figura 4.6), en caso contrario se mostrará en el header de la web (Figura 4.5).

El funcionamiento del buscador consiste en que cuando el usuario introduzca dos caracteres se empieza a mostrar el resultado de búsqueda, todo esto sin actualizar la página, ya que se hace uso de AJAX para ello. El usuario podrá seleccionar cualquiera de las películas mostradas para ver su detalle directamente o darle al botón de buscar y se mostrará una página con el listado de las películas que coincidan con el término de búsqueda.

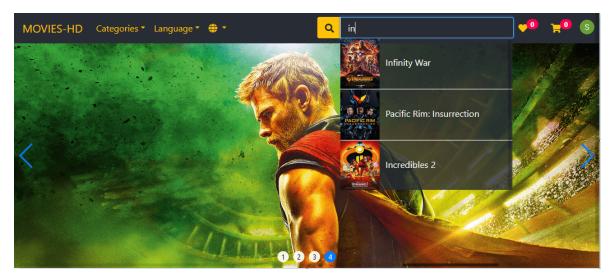


Figura 4.5: Buscador de película para dispositivos grandes

- Añadido el botón de búsqueda y el formulario de buscar películas por su nombre en el header de la web.
- Adaptado el formulario para cuando se acceda a la web desde un dispositivo móvil el buscador ocupe toda la pantalla.
- Importada la dependencia de jQuery UI para el autocompletado del buscador, así cuando el usuario escriba dos o más caracteres se empiecen a mostrar resultados.
- Creación del archivo *autocomplete-movies-search.js* para crear la vista del resultado de búsqueda mencionado en el punto anterior, el cual muestra una imagen de la película y su nombre. Si se accede a la web desde un ordenador, se podrá seleccionar la película con las flechas direccionales del teclado, en cualquier otro caso sería pulsando en la película.
- Añadido en el archivo *styles.css* las reglas de estilo para el formulario de búsqueda en pantalla completa y para el autocompletado de jQuery UI.
- Creación de la vista movies_searched.html para mostrar el resultado de búsqueda si el usuario no selecciona una película directamente del autocompletado y en vez de eso pusla el botón de búsqueda.
- Añadido el método de búsqueda de las películas por su nombre en las interfaces IMovieDao e IMovieService, por lo tanto, se ha implementado en MovieServiceImpl.

34 4.2. SPRINT 1

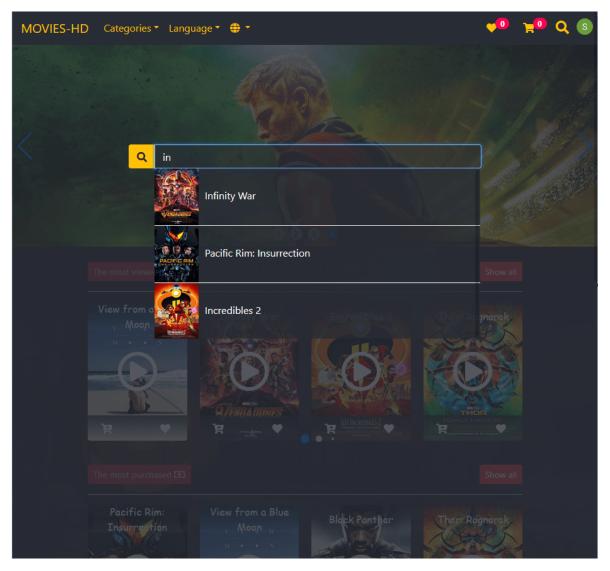


Figura 4.6: Buscador de película para dispositivos pequeños y medianos

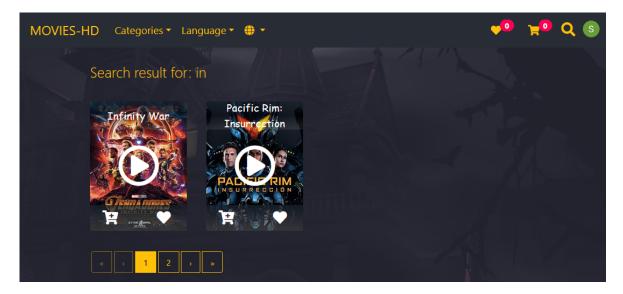


Figura 4.7: Resultado de búsqueda de películas

- Añadidas las siguientes URLs en MovieController para gestionar las peticiones de búsqueda de películas por sus nombres:
 - base-url/movies, el cual devuelve una lista de películas en formato JSON del resultado de buscar el término introducido por el usuario en el formulario, dicho resultado es el que se muestra en la (Figura 4.5 y Figura 4.6).
 - base-url/movies/searched, el cual se usa cuando el usuario pulsa el botón de buscar y se muestra una página con las películas que coincidan con el término de búsqueda.

4.2.4. HU 7. Filtro de búsqueda de películas

Esta historia de usuario se trata de poder filtrar la búsqueda de películas por el género de las mismas, al igual que por el idioma. Cada película puede aparecer en más de un filtro de búsqueda en cuanto al género, ya que pueden tener más de uno.

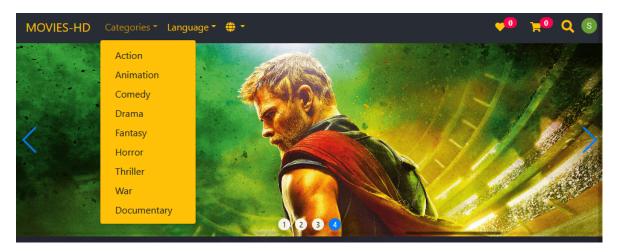


Figura 4.8: Filtro de búsqueda por categoría

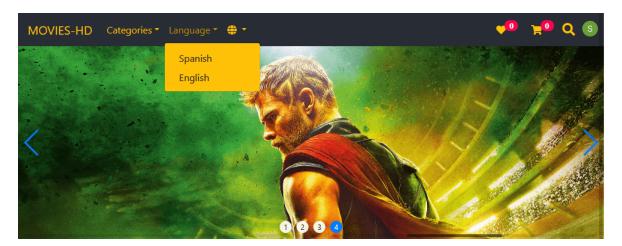


Figura 4.9: Filtro de búsqueda por idioma

- Creación la entidad Genre para almacenar los distintos géneros existentes que puede tener una película.
- Creación la entidad **MovieGenre** que sería la relación entre las entidades Movie y Genre.

36 4.2. SPRINT 1

 Añadido a la entidad Movie la relación @OneToMany con la entidad MovieGenre debido a que una película puede tener más de un género.

- Añadido a las interfaces IMovieDao e IMovieService los métodos para buscar películas por su género e idioma, por tanto, se ha creado también la implementación en MovieServiceImpl.
- Añadido en MovieController las siguientes URLs para poder filtrar la búsqueda de películas:
 - base-url/movies/genre/{name}, se obtiene de la base de datos las películas que coincidan con el género del valor del parámetro "name".
 - base-url/movies/language/{lang}, se obtiene de la base de datos las películas que tengan como audio el valor del parámetro "lang".
- Insertadas más películas en la base de datos.

4.2.5. HU 2. Sistema de inicio de sesión tradicional

En esta historia de usuario se implementa el sistema de inicio de sesión tradicional, el cual consiste en que el usuario inserte su correo electrónico y la contraseña para poder iniciar sesión en la web y así poder comprar películas y realizar otras funcionalidades.

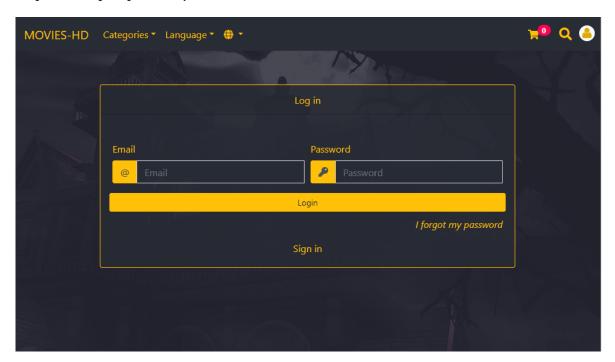


Figura 4.10: Formulario inicio sesión tradicional

- Añadidas las siguientes dependencias en el *pom.xml*:
 - spring-boot-starter-security para hacer el uso del sistema de seguridad y restringir el
 acceso a las diferentes páginas de la aplicación web.
 - *thymeleaf-extras-springsecrity4* para hacer uso de las opciones de seguridad en las vistas de la web haciendo uso del motor de plantilla Thymeleaf.
- Añadido al menú de opciones que se encuentra en el header de la web el enlace para mostrar la página con el formulario de inicio de sesión.

- Añadido en el enlace de inicio de sesión, la etiqueta de seguridad de Thymeleaf para mostrarlo solo cuando el usuario haya cerrado sesión con anterioridad.
- Creación de la página login.html que mostrará el formulario de inicio de sesión.
- Añadido en el archivo *styles.css* las reglas de estilo para el formulario.
- Creación del archivo form Validation.js para validar los campos del formulario e impedir que se envíen datos erróneos.
- Creación de la clase SpringSecurityConfig que será la encargada de añadir la seguridad a las diferentes URLs de la aplicación web, y así controlar el acceso
- Creación del controlador **LoginController** con la URL *base-url/login* para gestionar las peticiones de inicio de sesión y así mostrar mensajes al usuario del estado de la operación.
- Creación de la página *error_403.html* que mostrará el mensaje de error de acceso denegado cuando un usuario que no tenga permisos de acceso en ciertas páginas de la web.
- Creación de la clase **MvcConfig** para registrar la página *error_403.html*.
- Añadido en SpringSecurityConfig la configuración para mostrar la página de error de acceso denegado.
- Añadido en LoginController el manejo de error en el inicio de sesión.
- Añadido en el módulo DAO, la interfaz IUserDao para gestionar los usuarios en la base de datos. Añadido la declaración del método de buscar un usuario por su email.
- Añadido en el módulo services la interfaz IUserService y su implementación UserService-Impl para implementar el método de búsqueda de un usuario por su email en la base de datos.
- Creación de los siguientes elementos:
 - Entidad **User** para almacenar los datos de los usuarios, como el nombre, apellidos, correo, contraseña, entre otros.
 - Entidad Role que almacenará los tipos de roles: usuario y administrador en la aplicación web.
 - Servicio JpaUserDetailsService que será el encargado de cargar el usuario que inicie sesión con sus diferentes roles como un objeto usuario en el contexto de Spring Security y así controlar la sesión del mismo.
- Añadido en MvcConfig el bean BCryptPasswordEncoder para codificar las contraseñas de los usuarios. Un bean no es más que una anotación que se usa en Spring para el Inversion of Control (IoC) principle, permitiendo hacer uso del objeto al cual se le añade la anotación a través de la Dependency Injection (DI).
- Inyección de dependencia del servicio JpaUserDetailsService y el bean BCryptPasswordEncoder dentro de la clase SpringSecurityConfig para la configuración global de acceso a la aplicación.
- Insertadas más instrucciones SQL en el archivo import.sql para popular la base de datos con usuarios de prueba y sus roles.

38 4.2. SPRINT 1

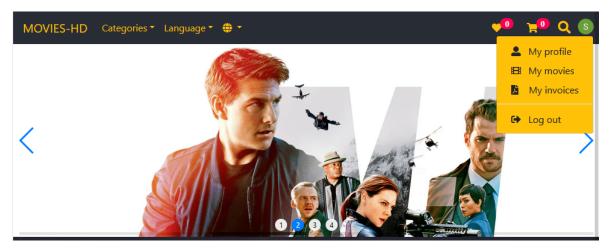


Figura 4.11: Formulario cierre de sesión

4.2.6. HU 5. Sistema de cierre de sesión

En esta historia de usuario se crea el enlace para que el usuario pueda cerrar la sesión de su cuenta. Listado de tareas de la historia de usuario:

- Añadido en el menú de opciones del header de la web, el enlace para cerrar la sesión.
- Añadido en el enlace de cierre de sesión, la etiqueta de seguridad de Thymeleaf para mostrar esta opción solo cuando el usuario haya iniciado sesión anteriormente.

4.2.7. HU 4. Sistema de registro

Esta historia de usuario consiste en la implementación de un formulario para el registro o creación de una cuenta de usuario en MoviesHD, ya que para poder comprar películas se debe de tener una cuenta. En el formulario de registro (Figura 4.12) se muestra los campos necesarios para el proceso de creación de una cuenta. En dicho formulario también se muestra el reCaptcha[31] de Google para evitar que programas informáticos realicen el registro de cuentas de usuarios falsas.

Para poder realizar el registro, el usuario tendrá que aceptar previamente los términos y condiciones de uso. Si un usuario ya existe, se le informará de ello y se le redirigirá a la página de inicio de sesión.

- Creación de la página *register.html* que mostrará el formulario de registro.
- Añadido en el archivo *styles.css* las reglas de estilo para el formulario.
- Creación del controlador UserController para gestionar las peticiones de los usuarios relacionas con la entidad User.
- Añadido en UserController la URL base-url/register para gestionar el registro de un usuario.
- Añadido en el formulario de inicio de sesión el enlace para registrarse.
- Añadido en las interfaces IUserDao e IUserService la declaración del método para registrar un nuevo usuario, al igual que su implementación en el servicio UserServiceImpl.
- Añadida en la configuración de seguridad de la clase SpringSecurityConfig la URL para el sistema de registro y así permitir a cualquier usuario el acceso al mismo.

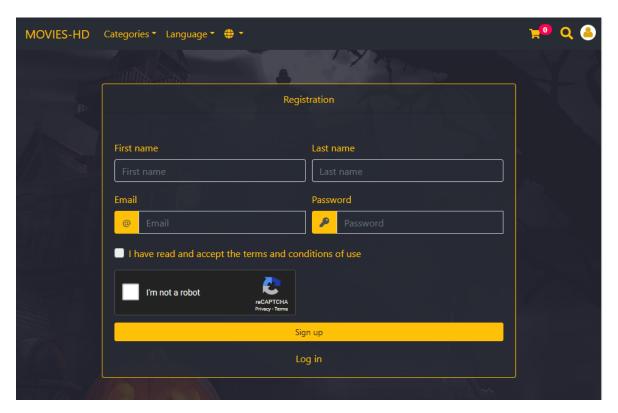


Figura 4.12: Formulario de registro

4.2.8. Retrospectiva del Sprint

La mayor parte de la interfaz gráfica de la web se ha hecho en este Sprint. Se han implementado las funcionalidades necesarias para que el usuario pueda hacer uso del portal web y poder ver el catálogo de películas con sus avances e información.

Las historias de usuarios a destacar en este Sprint son las de la página de inicio, debido a la estructura y diseño de las diferentes secciones de la misma, ya que al fin y al cabo es lo primero que ve el usuario cuando entra a la web. Por otro lado, el buscador de películas por su rápida respuesta a la hora de mostrar resultados y por su diseño responsivo que dependiendo del dispositivo y de su tamaño de pantalla se muestra con un diseño u otro.

En cuanto a dificultades en su implementación cabe destacar el buscador de películas, debido al aprendizaje de la librería de jQuery UI, para poder mostrar las imágenes de las películas en el resultado de búsqueda y el adaptar el diseño del buscador a diferentes dispositivos.

4.2.9. Mejoras del Sprint

- Añadidos los mensajes de error, información, advertencia y exitosos a mostrar al usuario como respuestas a ciertas tareas y acciones que puedan hacer en la web.
- Creación del módulo auth y el componente LoginSuccessHandler para mostrarle al usuario el mensaje de inicio de sesión satisfactorio.
- Creación del controlador ErrorPageController al igual que la página error.html para gestionar y mostrar los errores ajenos al usuario, como pueden ser el Error 404 Page Not Found, Error 500 Internal Server Error, entre otros.
- Añadido el reCaptcha de Google en el formulario de registro o creación de una cuenta de usuario.

4.3. SPRINT 2

 Desplegado de la página web en el dominio público www.movies-hd.cf mediante los siguientes pasos:

- Creación de una cuenta en Freenom[32] para el registro del dominio público.
- Creación de una Hosted Zone en AWS Route 53 para el servicio de DNS.
- Incluido los DNS creados en el punto anterior en el domino público creado en Freenom.
- Creación con AWS Certificate Manager un certificado de capa de conexión segura/seguridad de la capa de transporte (SSL/TLS) para el domino.

4.2.10. Configuraciones extras del proyecto

- Creación de los Spring Profile: Development y Production para trabajar en el desarrollo del software en local y en producción.
- Creación de los siguientes archivos de configuración:
 - *application-dev.properties*, el cual tendrá las configuraciones necesarias solo para el perfil de desarrollo.
 - *application-prod.properties*, el cual tendrá las configuraciones necesarias solo para el perfil de producción.
 - DevSpringSecurityConfig para la configuración de seguridad de la web en el perfil de desarrollo.
 - **ProductionSpringSecurityConfig** para la configuración de seguridad de la web en el perfil de producción.
- Modificación del componente de configuración MvcConfig, añadiendo el bean DataSource para la configuración de la conexión con la base de datos en local para el perfil de desarrollo.
- Modificación del archivo de configuración application.properties, añadiendo las configuraciones que son comunes para los perfiles de desarrollo y producción.

4.3. **SPRINT 2**



Figura 4.13: Historias de usuario del Sprint 2

A continuación, se pasa a detallar el desarrollo de cada historia de usuario (HU).

4.3.1. HU 12. Carro de la compra

En esta historia de usuario a cada película se le ha añadido un icono con forma de cesta de compra para que cuando el usuario lo pulse se añada dicha película al carro de la compra. Al pulsar el icono de la cesta, se mostrará una notificación (Snackbar, Figura 4.14). Una vez que se añada una película al carro de la compra, ya no se podrá volver añadir, esto es debido a que solo se puede comprar la película una sola vez.

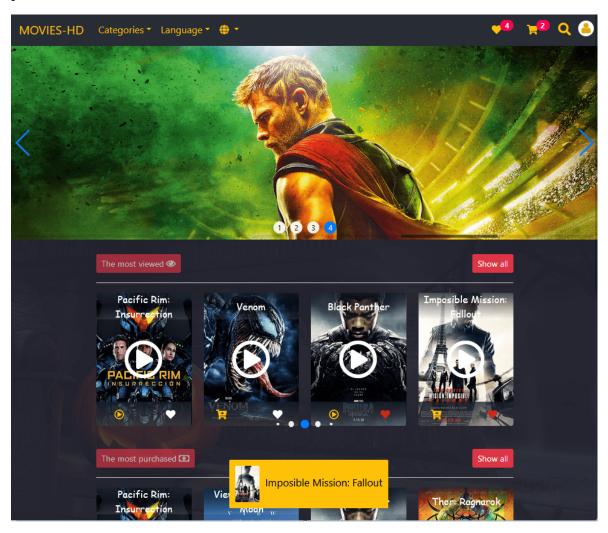


Figura 4.14: Notificación de añadir película al carro de la compra

- Creación de los siguientes elementos:
 - Módulo utils, para organizar las diferentes clases que se vayan necesitando en el desarrollo del proyecto y que serían de utilidad.
 - Clase **Cart** que almacenará la cantidad total de elementos que tiene el carro de la compra y la película añadida en ese momento.
 - Controlador CartController para la gestión del carro de la compra, el cual se almacenará en la sesión del usuario. La URL añadida para esta historia de usuario es base-url/cart/addmovie, que como su nombre indica es para añadir una película al carro de la compra.
 - Archivo ajax.js para gestionar las peticiones del usuario respecto al carro de la compra, el cual está hecho con AJAX para una mejor experiencia de uso. Aquí también se crea el SnackBar mencionado con anterioridad.

42 4.3. SPRINT 2

- Incluido el archivo *ajax.js* en el footer de la web.
- Añadidos todos los enlaces en el icono de la cesta de compra de todas las imágenes de las películas en las distintas páginas de la aplicación web.
- Modificada la lógica de la vista de las películas donde aparece el icono de la cesta de la compra, para que cuando dicha película este añadida al carro, el icono esté de color amarillo y no se pueda volver añadir y en caso contrario se encuentre de color blanco.
- Incluidas las reglas de estilo del SnackBar en el archivo styles.css.

4.3.2. HU 13. Detalle carro de la compra

La historia de usuario consiste en mostrar el detalle del carro de la compra, y así poder ver el precio total del mismo, pudiendo eliminar las películas de forma individual o vaciar el carro directamente.

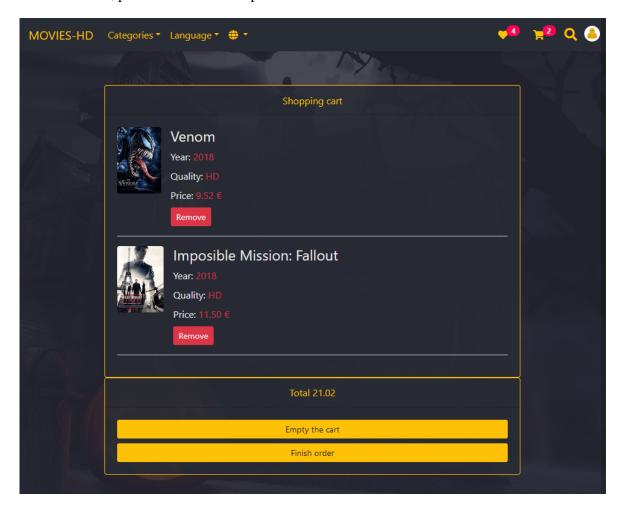


Figura 4.15: Detalle del carro de la compra

- Añadido el enlace de mostrar el detalle del carro de la compra en la header de la web.
- Creación de la página cart_detail.html, que como su nombre indica, será para mostrar el detalle del carro de la compra.
- Incluidas las siguientes URLs en CartController:

- base-url/cart/all, el cual mostrara el detalle del carro de la compra con todas las películas añadidas.
- base-url/cart/remove-movie, para poder eliminar una determinada película del carro de la compra.
- base-url/cart/empty, para vaciar el carro de la compra.
- Añadido al archivo *ajax.js* los métodos de eliminar una película del carro de la compra, al igual que el de vaciarlo por completo.

4.3.3. HU 14. Pasarela de pago

Esta historia de usuario le permite al usuario realizar la compra de películas mediante una cuenta de PayPal o una tarjeta de crédito/debito. Aquí se hace uso de la API de Braintree para la gestión de todo el proceso de la pasarela de pago, tanto desde la interfaz de gráfica, como de la lógica en el servidor.

Las películas solo se pueden comprar una vez, por tanto, se han añadido ciertas lógicas, tanto en el servidor, como en las vistas de las películas en la web para implementar dicho comportamiento. Una vez efectuado el pago de forma satisfactoria, se procede a eliminar las películas del carro de la compra.

Los usuarios que realicen la compra de películas mediante PayPal también podrán ver el detalle de dicha compra en su cuenta de PayPal (Figura 4.22).

Por motivos legales, la opción de pago no está disponible al público. Esto se debe a que se tendría que ralizar todos los trámites legales para la creación de un comercio electrónico real, al igual que los contratos con las distribuidoras de contenido cinematográfico. En el proyecto se ha creado un usuario con el Rol de Administrador para testear la plataforma de pago.

En un futuro, realizando los trámites mencionados en el párrafo anterior, si se habilitaría la opción de pago en el Portal Web, es por ello que se ha añadido como posibles extensiones del proyecto.

- Creación de una cuenta **PayPal Sandbox**[33], y así poder crear usuarios de prueba para el uso de la API de Braintree en la pasarela de pago.
- Creación de una cuenta en Braintree para el uso de su API en la pasarela de pago y configurar los datos de la cuenta de PayPal creada en el paso anterior.
- Añadida la dependencia *braintree-java* en el *pom.xml* y así poder hacer uso de la API de Braintree en el lado del servidor.
- Creación del módulo config dentro de los recursos web del proyecto para almacenar archivos de configuración.
- Creación dentro del módulo config el archivo config.properties con las variables de configuración de la API de Braintree.
- Creación del módulo **BraintreeGateway** para estructurar los elementos necesarios de la pasarela de pago.
- Creación de la clase BraintreeGatewayFactory con los métodos necesarios para cargar la configuración de Braintree almacenadas en el archivo config.properties.
- Modificada la clase principal del proyecto MovieshdApplication para cargar la configuración de Braintree cuando se inicie la ejecución de la aplicación.

4.3. SPRINT 2

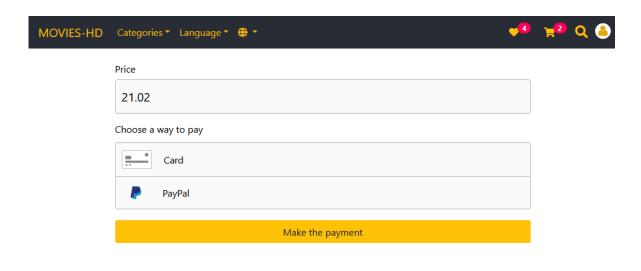


Figura 4.16: Pasarela de pago, proceso de selección del método de pago

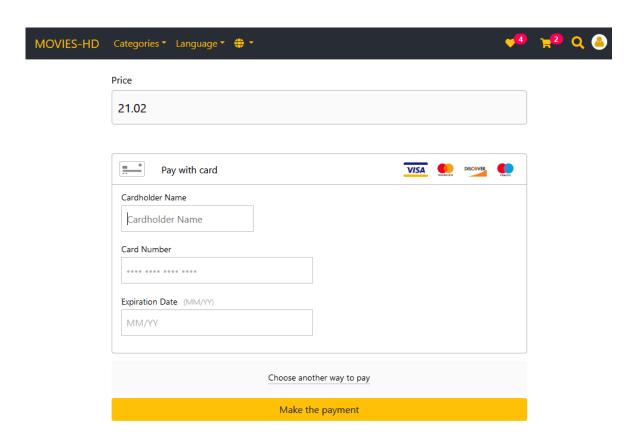


Figura 4.17: Pasarela de pago, compra mediante tarjeta

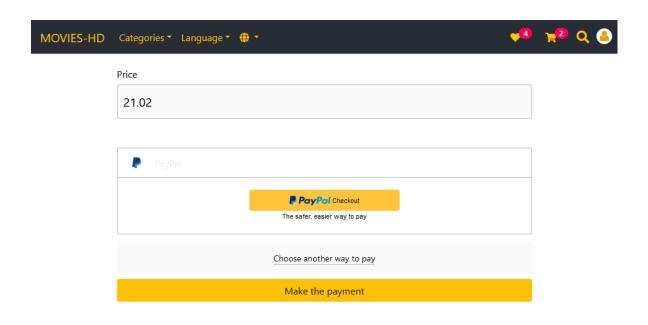


Figura 4.18: Pasarela de pago, compra mediante PayPal

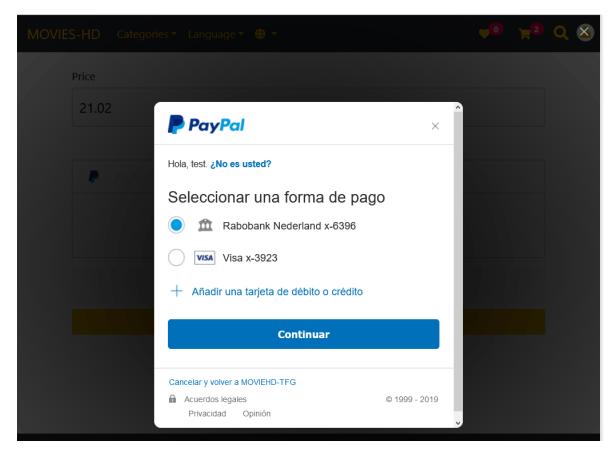


Figura 4.19: Pasarela de pago, compra con la cuenta de PayPal

4.3. SPRINT 2

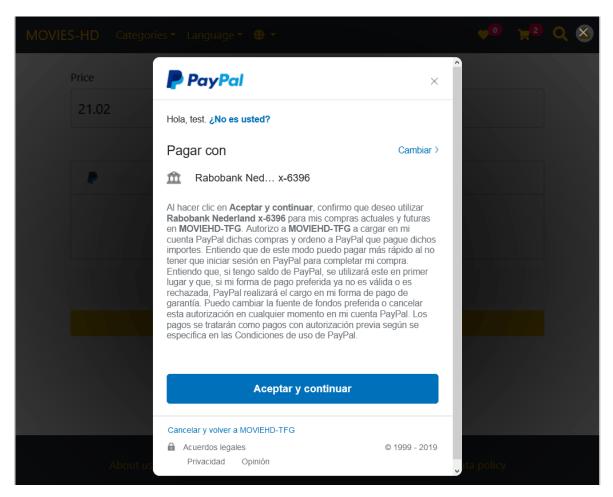


Figura 4.20: Pasarela de pago, aceptación de condiciones de pago con PayPal

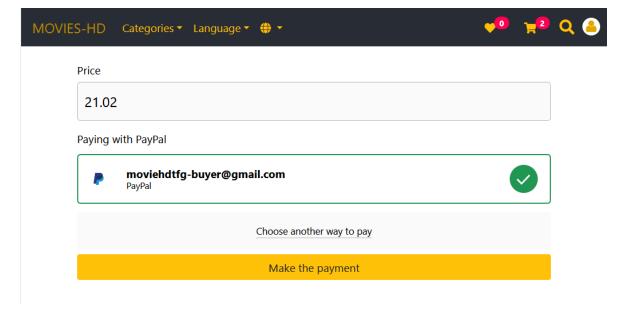


Figura 4.21: Pasarela de pago, realización de compra mediante PayPal



3 may 2019 19:14:41 CEST Id. de transacción: 75M212170B0503913

Estimado(a) test buyer:

Ha enviado un pago por valor de €21,02 EUR a test facilitator's Test Store (moviehdtfg-facilitator@gmail.com)

Esta transacción puede tardar unos instantes en aparecer en su cuenta.

Vendedor test facilitator's Test Store moviehdtfg-facilitator@gmail.com

Instrucciones para el vendedor No ha introducido instrucciones.

Descripción	Precio por unidad	Cantidad	Importe
Venom	8,90 EUR	1	€8,90 EUR
Imposible Mission: Fallout	10,75 EUR	1	€10,75 EUR
		Subtotal Impuesto:	€19,65 EUR €1,37 EUR
		Total	€21,02 EUR
		Pago	€21,02 EUR

Pago enviado a moviehdtfg-facilitator@gmail.com

Figura 4.22: Pasarela de pago, detalle de la compra mediante PayPal

- Creación dentro del módulo utils la clase SessionUtil para obtener los objetos de la sesión, en este caso las películas que se vayan añadiendo en el carro de la compra. También se ha añadido la lógica para que cuando un usuario inicie sesión se muestre en cada película comprada por él el icono del reproductor de la misma, en vez del icono de la cesta de la compra.
- Modificada la lógica de la vista de las películas donde aparece el icono de la cesta de la compra, para que cuando dicha película haya sido comprada por el usuario, el icono que aparezca sea el del reproductor, evitando así que el usuario vuelva a comprar la película y en caso contrario se muestre el icono de la cesta de la compra.
- Creación del controlador BraintreeGatewayController con la URL base-url/checkouts, que es donde se encuentra toda la lógica del proceso de pago, ya sea desde obtener las películas del carro de la compra, hacer uso de la API de Braintree para la creación de la transacción del cobro, el almacenamiento en la base de datos del detalle de la compra y de la información de pago.

48 4.3. SPRINT 2

Creación de la página new.html que mostrará el formulario de pago, haciendo uso de Drop-in
 UI de Braintree[34], que no es más que la interfaz gráfica de la pasarela de pago (Figura 4.16).

- Incluidas las reglas de estilo de la pasarela de pago en el archivo styles.css.
- Implementación de Spring Security con anotaciones en vez de lo hecho anteriormente para una mejor gestión de la seguridad.
- Restricción del acceso a la pasarela de pago, permitiendo el uso solo a los usuarios con roles de administrador.

4.3.4. HU 15. Confirmación de compra

En esta historia de usuario se crea la funcionalidad del envío de un correo electrónico al usuario y de forma automática con la confirmación de la compra. El correo enviado contiene el detalle de la compra, ya sea las películas compradas, con su nombre, imagen y precio individual, al igual que el precio total de la compra y el identificador de la orden.

El diseño del correo electrónico está hecho para que se vea de la misma forma en los proveedores de servicios de correos actuales.

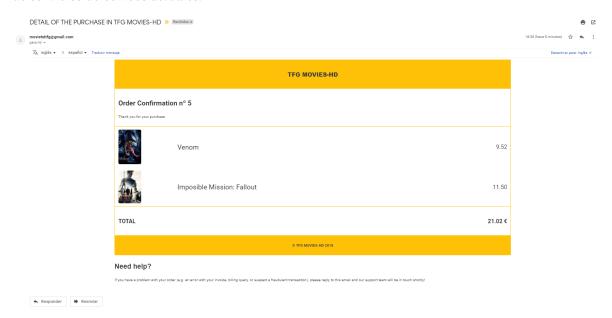


Figura 4.23: Confirmación por correo electrónico de la compra

- Incluida la dependencia spring-boot-starter-email en el pom.xml para el envío de correos electrónicos.
- Creación de una contraseña de aplicación para introducirla en una de las variables incluidas en el paso anterior. Una contraseña de aplicación te permite iniciar sesión en tu cuenta de Google desde aplicaciones instaladas y así no se necesita poner la contraseña de acceso real de la cuenta de correo.
- Incluidas en el archivo *application.properties* las variables de configuración para el envío de correos electrónicos haciendo uso el proveedor de correo de Google.
- Creación del módulo email para estructurar las clases necesarias en el desarrollo del envío de correos electrónicos.

- Creación del módulo Model y la clase Mail que almacenará los objetos básicos de un correo electrónico, por ejemplo, el asunto, el destinatario, el contenido, etc.
- Añadida la anotación @EnableAsync para hacer uso de concurrencia a la hora del envío del correo electrónico y así el usuario pueda seguir interactuando con la web.
- Creación del servicio MailClientBuilder, que tendrá los métodos necesarios para el envío de los correos electrónicos.
- Creación de la plantilla *payment_confirmation.html*, que no es más que una página web con la estructura y diseño del correo electrónico a enviar.
- Añadido el servicio MailClientBuilder en BraintreeGatewayController, al igual que el código necesario para enviar el correo electrónico de confirmación de la compra una vez que el usuario realice la transacción.

4.3.5. HU 19. Acceso películas compradas

En esta historia de usuario se ha añadido en el menú de opciones del header un enlace para que el usuario pueda ver sus películas compradas. Se ha creado una página donde se muestra el listado de las películas compradas por el usuario y poder visualizarlas en cualquier momento del día sin ningún tipo de restricción. En la sección de películas compradas el usuario puede añadir o eliminar las películas a su lista de favoritos de la misma forma que lo haría en las otras páginas de la web.

En cuanto a la reproducción de las películas, se ha usado el reproductor multimedia Plyr (Figura 4.27), el cual es un proyecto con licencia del Massachusetts Institute of Technology (MIT). Plyr aparte de tener las opciones generales de un reproductor, te permite seleccionar el idioma del subtítulo que pueda tener una película.

En cuanto al diseño del reproductor Plyr se ha adaptado a los colores de la aplicación web MoviesHD. Por otro lado, se ha añadido una funcionalidad interesante que suele verse en pocos reproductores de vídeos como por ejemplo YouTube, la cual consiste en mostrar imágenes en miniaturas (Thumbnails) mientras se pasa el ratón por encima de la línea de tiempo de la película. Estas thumbnails muestran la imagen que coincide con la línea de tiempo de la película en ese momento. Para el uso de esta funcionalidad hay que crear lo que se conoce como Sprite (Figura 4.25), el cual consiste en un conjunto de imágenes combinadas en un solo archivo. Por otro lado, se crea un archivo The Web Video Text Tracks Format (WEBVTT) que son como los de los archivos de subtítulos de las películas, pero con otra estructura donde se almacena las coordenadas "x" e "y", al igual que el ancho y alto de las imágenes dentro del Sprite en cada rango de tiempo de la película. Para la creación de los archivos Sprite y WEBVTT se hace uso del servicio web Videohop[35].

```
WEBVTT

1
00:00:00.000.000 --> 00:00:01.000
240p-00001.jpg#xywh=0,0,577,240

2
00:00:01.000 --> 00:00:02.000
240p-00001.jpg#xywh=577,0,577,240
```

Figura 4.24: Archivo WEBVTT para el uso de las imágenes en miniaturas

50 4.3. SPRINT 2

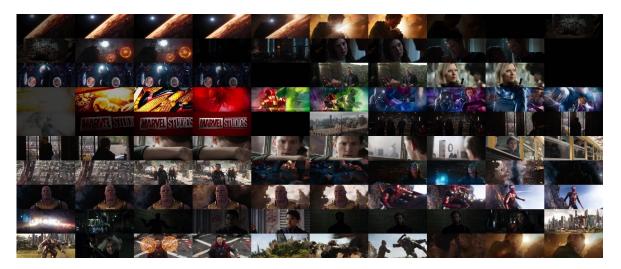


Figura 4.25: Sprite, conjunto de imágenes en miniaturas

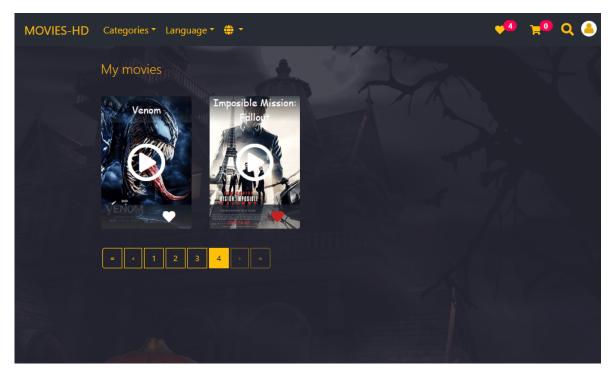


Figura 4.26: Listado de películas compradas

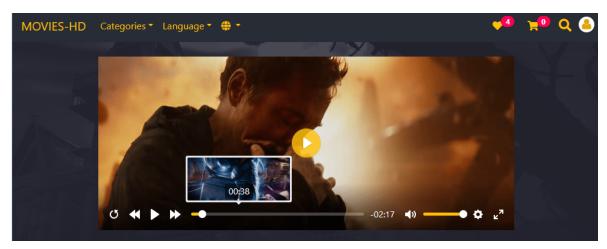


Figura 4.27: Reproductor multimedia Plyr

- Creación de los siguientes elementos:
 - Entidad InvoiceItem, la cual almacenará la película comprada y la cantidad de la misma, que en este caso siempre será uno ya que, al ser películas en formato digital, no tiene sentido comprar más de una.
 - Entidad **Invoice**, donde se almacenará la fecha de creación de la factura, la entidad User que será el usuario que realizó la compra, el precio total, etc.
 - Añadida en la entidad Invoice la relación @OneToMany con la entidad InvoiceItem.
 - Interfaz **InvoiceDao** para el acceso a la entidad Invoice, la cual proporcionará la estructura de los métodos CRUD aparte de los creados en un futuro si la aplicación web lo requiere.
 - Interfaz InvoiceService la cual contendrá la estructura de todos los métodos para la entidad Invoice, ya sea el CRUD y otros que se vayan necesitando en el desarrollo del proyecto.
 - Clase InvoiceServiceImpl, la cual contiene la implementación de todos los métodos declarados en la interfaz InvoiceService.
- Incluidos los nuevos elementos en BraintreeGatewayController para almacenar la transacción de la compra en la base de datos.
- Incluida la estructura del método para obtener las facturas con sus películas en las interfaces IUserDao e IUserService.
- Implementado el método incluido en el punto anterior en la clase UserServiceImpl para que el usuario puede tener acceso a sus películas compradas.
- Insertado más instrucciones SQL en el archivo import.sql para popular la base de datos con facturas y sus detalles.
- Añadido en el menú de opciones del header de la web el enlace para mostrar las películas compradas, para ello el usuario debe haber iniciado sesión con anterioridad.
- Creación de la *página movies.html* para mostrar las películas compradas.
- Añadidas las siguientes URLs en UserController para la reproducción de las películas:
 - base-url/my-movies, para obtener de la base de datos las películas compradas por un usuario.
 - base-url/watch, para generar el token para la reproducción de la película.
 - base-url/watch/{token}, para reproducir la película con el token generado en la URL anterior.
- Añadidas las siguientes dependencias en el *pom.xml*:
 - jjwt para la generación y validación de tokens con JWT.
 - **jets3t**[36], conjunto de herramientas Java de código abierto y gratuito para Amazon S3 y Amazon CloudFront.
 - spring-mobile-starter, para el uso del Framework Spring Mobile.
- Creación de los siguientes componentes:
 - Módulo **jwt** para estructurar las clases necesarias para la creación de tokens.

52 4.3. SPRINT 2

 Enumeración JwtSecretType para guardar los tipos de codificación en la creación de tokens.

- Clase **JwtFactory**, que tendrá los métodos para generar y validar los tokens.
- Módulo aws, para estructurar las clases necesarias para las firmas de las URLs de las películas que se encuentran en los buckets de Amazon S3.
- Clase **SignedURL**, que contiene la lógica para firmar la URL de la película a reproducir, evitando así que nadie más tenga acceso a dicha película.
- Creación del par de claves de CloudFront para firmar URLs. Esto se hace a través de la consola de Amazon, una vez creada se podrá descargar la clave privada con nombre pk-APKAJZWCQHG5TBGX2HLQ.pem en este caso. Debido a que la clave privada generada tiene la extensión .pem, se tiene que convertir a la extensión .der para poder usarla en el lenguaje de Java.

Para ello se hace uso OpenSSL[37] ejecutando el siguiente comando: openssl pkcs8 -topk8 - nocrypt -in pk-APKAJZWCQHG5TBGX2HLQ.pem -inform PEM -out pk-APKAJZWCQHG5TBGX2HLQ.der -outform DER

- Añadida la clave privada *pk-APKAJZWCQHG5TBGX2HLQ.der* dentro del módulo config en los recursos web del proyecto.
- Creación de la página *player.html* para mostrar el reproductor de vídeo con la película a reproducir.
- Añadida la hoja de estilo del reproductor Plyr en el head de la web.
- Incluidas las reglas de estilo propias para el reproductor Plyr en el archivo styles.css.
- Incluida la librería javascript del reproductor Plyr en el footer de la página player.html.
- Añadido el enlace en el icono de reproducir la película en las vistas de las mismas.
- Añadido los beans necesarios en la clase de configuración MvcConfig para el uso del framework Spring Mobile y así poder identificar si el usuario accede desde un dispositivo móvil o tableta y poder mostrar ciertas opciones en el reproductor Plyr.

4.3.6. HU 18. Historial de compras

En esta historia de usuario se ha creado la funcionalidad de que un usuario tenga acceso a todas sus facturas (Figura 4.28) con sus detalles (Figura 4.29) y donde pueda generar y descargar las mismas en formato pdf (Figura 4.30). La factura generada contendrá su identificador, fecha de la compra, el total de la factura, los datos del usuario, los datos de la información de pago y el listado de películas compradas con sus correspondientes precios de compra.

- Añadido el enlace en el menú de opciones dentro del header de la web para mostrar el listado de facturas del usuario, para ello debe de haber iniciado sesión previamente.
- Creación de las siguientes páginas:
 - invoice.html que mostrará una tabla con el listado de todas las facturas del usuario.
 - invoice_detail.html, la cual mostrará el listado de películas compradas de una factura determinada.
- Incluidas las reglas de estilos en el archivo styles.css de las páginas creadas en el punto anterior.

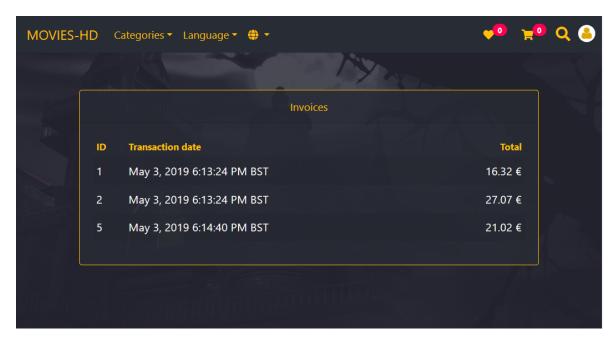


Figura 4.28: Listado de compra

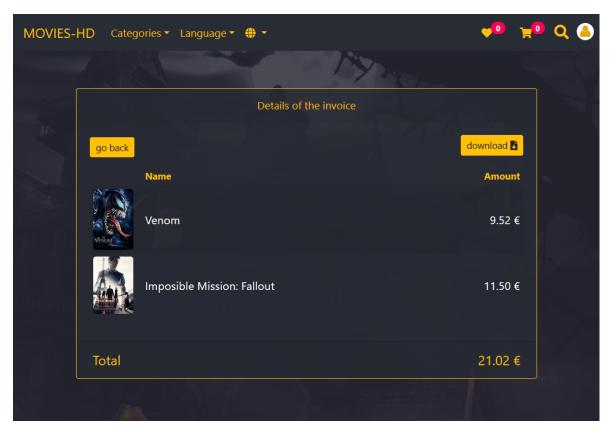


Figura 4.29: Detalle de la compra

54 4.3. SPRINT 2

TFG MOVIES-HD

Ctra. Tafira Baja A 35017

Las Palmas de Gran Canaria

España



Invoice information

Identifier 5

Date May 3, 2019 6:14:40 PM BST

Amount 21.02 €

Holder information

Full name Sergio López Díaz
Email sldiaz04@gmail.com
Phone +34123456789
Address Colombia

Postal code 35010

City Las Palmas de Gran Canaria

Country España

Payment information

Payment method PAYPAL

Buyer's email moviehdtfg-buyer@gmail.com

Buyer's ID Y3SJTK3YME5ZQ

First name test Last name buyer

Payment identifier PAYID-LTGHOAA3UV60792EL6820939

Items

Title	Quantity	Amount per unit	Total amount
Venom	1	8.90	8.90€
Imposible Mission: Fallout	1	10.75	10.75€
		Subtotal	19.65€
		Taxes 7%	1.37 €
		Total	21.02€

Figura 4.30: Factura de la compra en formato pdf

- Añadida la estructura del método en las interfaces IUserDao e IUserService para obtener las facturas del usuario.
- Implementado el método creado en el punto anterior dentro de la clase UserServiceImpl.
- Añadida la estructura del método en las interfaces InvoiceDao e InvoiceService para obtener el detalle de una factura determinada correspondiente a un usuario.

- Implementado el método creado en el punto anterior dentro de la clase InvoiceServiceImpl.
- Creación en UserController de la URL base-url/my-invoices para obtener las facturas del usuario.
- Creación del controlador InvoiceController con la URL base-url/invoice/{id} para obtener el detalle de una factura a través de su identificador.
- Insertado código javascript dentro de la página invoice.html para capturar el evento "onclick" de cada fila de la tabla y así mostrar el detalle de cada factura.
- Incluidas las variables de configuración en el archivo *application.properties* para la generación de la factura en formato pdf.
- Añadida la dependencia **openpdf**[38] en el *pom.xml* para la creación de la factura en el lado del servidor en formato pdf.
- Creación el módulo view y el componente InvoicePdfView para gestionar la petición del usuario de generar la factura en formato pdf.
- Incluido el enlace de descargar factura en la vista de detalle de la misma.

4.3.7. HU 17. Recuperar contraseña

Esta historia de usuario consiste en la implementación de la funcionalidad que le permita al usuario recuperar el acceso a su cuenta mediante el cambio de contraseña. En el formulario de inicio de sesión se ha añadido un enlace para que el usuario al pulsarlo se le redirija a otra página (Figura 4.31) donde se le pide que ingrese su correo electrónico, si no existe el usuario con dicho correo, se le informará de ello. El formulario para introducir el correo electrónico tiene el reCaptcha de Google para evitar que programas informáticos realicen esta operación.

Una vez que se comprueba que el usuario existe, se le enviará un correo electrónico (Figura 4.32) con las indicaciones para el cambio de la contraseña. Dentro de esas indicaciones hay un enlace, el cual tiene una hora de validez por cuestiones de seguridad y en caso de que dicho período se supere, se tendrá que volver a solicitar el envío del correo electrónico. Cuando el usuario pinche el enlace se le redirigirá a otro formulario (Figura 4.33) donde se le pedirá que introduzca la nueva contraseña dos veces para que esté seguro que no haya cometido un error al escribirla, si las contraseñas no coinciden se le indicará de ello.

Por otra parte, cuando el usuario cambie la contraseña se le enviará otro correo electrónico (Figura 4.34) con la confirmación del cambio, así se mantiene al usuario informado del cambio de la contraseña por motivos de seguridad.

- Añadido en el formulario de inicio de sesión el enlace para restablecer la contraseña.
- Creación de las siguientes páginas y plantillas web:
 - email_form.html, página donde se le muestra al usuario un formulario para que introduzca su correo electrónico y así se le envíe las instrucciones para restablecer la contraseña de acceso.
 - retrive_password.html, plantilla del correo electrónico que se le envía al usuario con toda la información necesaria para restablecer la contraseña.
 - retrive_form.html, página donde se le muestra al usuario un formulario para poder cambiar la contraseña de acceso.

56 4.3. SPRINT 2

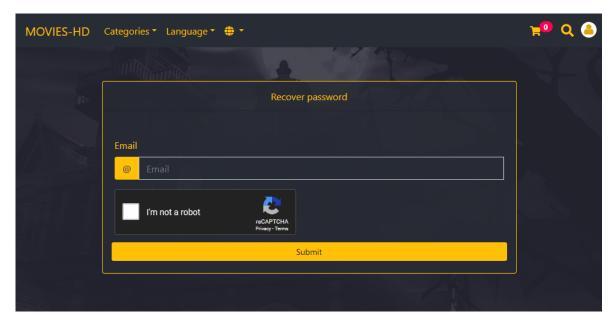


Figura 4.31: Formulario de envío del correo electrónico de recuperación de contraseña

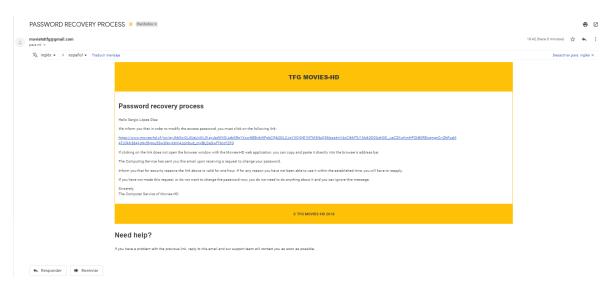


Figura 4.32: Correo electrónico del proceso de recuperación contraseña

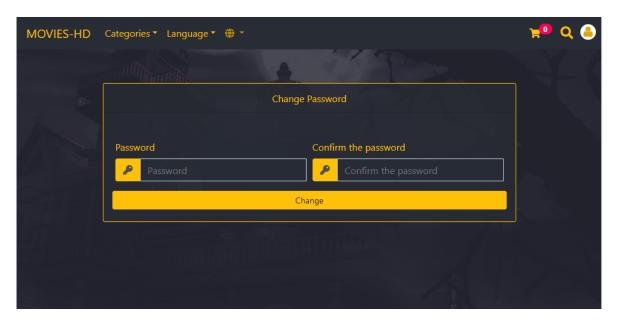


Figura 4.33: Formulario para el cambio de contraseña

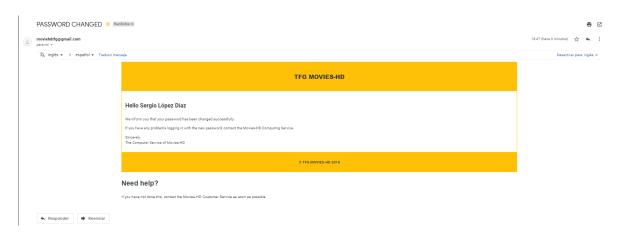


Figura 4.34: Correo electrónico de confirmación del cambio de contraseña

 password_changed.html, plantilla del correo electrónico que se le envía al usuario una vez haya cambiado la contraseña de acceso.

- Incluidas las reglas de estilos de las páginas creadas en el paso anterior en el archivo styles.css.
- Creación del controlador JwtController con las siguientes URLs para el proceso de recuperación de la contraseña de acceso:
 - base-url/jwt/send-email, para mostrarle al usuario la página con el formulario para el envío del correo electrónico.
 - *base-url/jwt/recipient-email*, donde se implementa toda la lógica para el envío del correo electrónico con las instrucciones para el cambio de la contraseña.
 - base-url/jwt/{token}, donde se implementa la lógica para comprobar que el enlace para restablecer la contraseña sea válido y si ese fuera el caso se le mostraría al usuario otra página con el formulario para el cambio de la contraseña.
 - base-url/jwt/change-password, donde se implementa la lógica para el cambio de la contraseña y el envío del correo electrónico con la confirmación del cambio.
- Añadidas las URLs creadas en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario el acceso a la funcionalidad del restablecimiento de la contraseña.

4.3.8. Retrospectiva del Sprint

Con la implementación de este Sprint, el usuario ahora puede realizar la compra de películas, ver el detalle de sus facturas y descargarlas en formato pdf. También puede ver las películas compradas en cualquier momento, al igual que recuperar el acceso a su cuenta de usuario.

Son muchas las historias de usuarios a destacar en este Sprint. Dentro de las más importantes se encuentra la pasarela de pago, ya que es donde se implementa todo el proceso de cobro de la aplicación web. Otra historia de usuario a destacar es la de confirmación de compra, en la cual se crea y se envía un correo electrónico con un diseño personalizado al usuario una vez haya realizado la operación. Por otro lado, la historia de usuario de acceso a películas compradas, principalmente el reproductor multimedia Plyr y las funcionalidades extras como las imágenes en miniaturas. Por último, la historia de usuario de recuperar contraseña, por la creación de tokens con JWT para la URL de restablecimiento de la contraseña y la creación y envío de varios correos electrónicos.

En cuanto a dificultades, cabe destacar principalmente la pasarela de pago, por su complejidad en el proceso de creación de la transacción en el servidor, la extensa lectura y comprensión de la documentación de la API de Braintree para llevar a cabo dicha historia de usuario, inclusive se ha tenido que contactar con el soporte técnico de Braintree para realizar algunas consultas en cuanto a los diferentes niveles de la transacción y así poder generar el detalle de la compra cuando el usuario realiza el pago con una cuenta de PayPal. Por otro lado, la funcionalidad de las imágenes en miniaturas para el reproductor Plyr, debido al proceso de creación del conjunto de imágenes y el archivo WEBVTT para poder hacer uso de esas imágenes en el reproductor. Por último, la creación de las plantillas de los diferentes correos electrónicos enviados, debido a la adaptación del diseño para su correcto funcionamiento en los diferentes proveedores de servicios de correo.

4.3.9. Mejoras del Sprint

- Añadido el reCaptcha de Google en el formulario de recuperación de la contraseña.
- Agregadas las imágenes en miniaturas en el reproductor de vídeo Plyr para que cuando pasemos el ratón sobre la barra de tiempo se muestren la imagen de la película en ese instante de tiempo.

- Creación de la enumeración PaymentType y la entidad PaymentInformation para almacenar en la base de datos la información de pago, al igual que mostrarla en la factura.
- Añadido en el icono de reproducir de cada película comprada un enlace para la reproducción de la misma de forma directa sin tener que ir a la sección de "mis películas".



Figura 4.35: Historias de usuario del Sprint 3

A continuación, se pasa a detallar el desarrollo de cada historia de usuario (HU).

4.4.1. HU 3. Sistema de inicio de sesión con Google o Facebook

En esta historia de usuario se hace uso de la API de Firebase para añadir la funcionalidad a la web de poder iniciar sesión con cuentas de proveedores de servicios como Google o Facebook. Con esta funcionalidad se le da la opción al usuario de crear una cuenta sin la necesidad de rellenar un formulario de registro.

Los datos a los que se accede a través de esos proveedores de servicios son los básicos del perfil del usuario, por ejemplo, el nombre, apellidos, correo electrónico e imagen de perfil, todo ello más que suficiente para crear una cuenta en MoviesHD. La primera vez que se inicie sesión con Facebook o Google, es donde se creará la cuenta de usuario, en las siguientes solo se iniciará la sesión.

El usuario que cree su cuenta a través de algunos de los proveedores de servicios, luego podrá iniciar sesión a través de la forma tradicional, mediante el correo electrónico que tenga asociado a su cuenta de proveedor de servicio y una contraseña. La contraseña la tendrá que cambiar, por lo menos una vez en su cuenta de perfil de MoviesHD mediante el acceso con algunos de los proveedores de servicios, ya luego podrá acceder a su cuenta mediante cualquiera de los métodos disponibles.

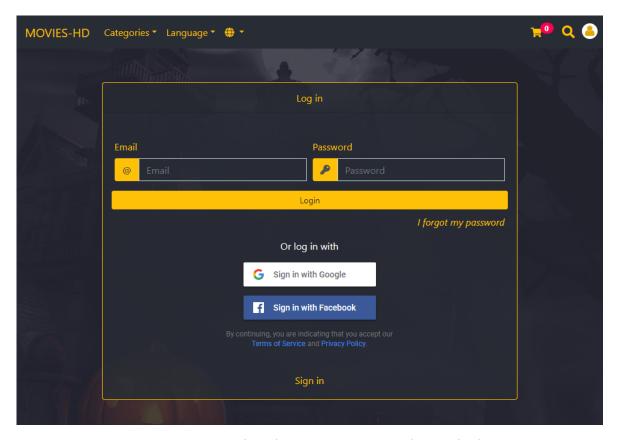


Figura 4.36: Formulario de inicio sesión con Google o Facebook

Listado de tareas de la historia de usuario:

- Creación de un proyecto en la consola de Firebase para su uso en esta historia de usuario.
- Creación de las claves de las APIs de Facebook y Google para su uso en el proyecto de Firebase creado en el punto anterior.
- Añadida la hoja de estilos de **FirebaseUI** en el head de la web.
- Generación de una clave privada en la cuenta de servicio del proyecto creado en el primer punto y descargado el archivo en formato JSON con las credenciales de la cuenta para el uso del Software Development Kit (SDK) Firebase Admin[39] en el servidor.
- Añadido el archivo JSON con las credenciales de la cuenta en el módulo config dentro de los recursos web del proyecto.
- Añadida la dependencia **firebase-admin** en el *pom.xml.*
- Creación de los siguientes elementos:
 - Módulo **firebase**, para estructurar los elementos necesarios en el desarrollo de la autenticación con Firebase.
 - Clase de configuración **FirebaseConfig**, para cargar las credenciales de Firebase al inicio de la ejecución del proyecto.
 - Controlador FirebaseController con la URL base-url/firebase/idToken para cuando el
 usuario inicie sesión con el sistema de Firebase enviar el idToken generado al servidor y
 obtener los datos de usuario de ese proveedor de servicio y así crear la cuenta de usuario.

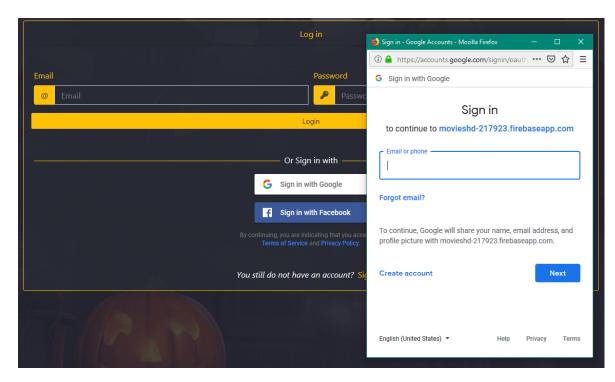


Figura 4.37: Inicio de sesión con Google

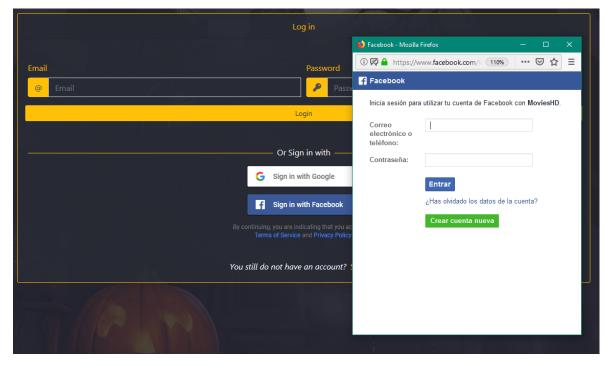


Figura 4.38: Inicio de sesión con Facebook

 Añadida la URL creada en el punto anterior en la configuración de seguridad de DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario el acceso o creación de su cuenta de usuario mediante Facebook o Google.

- Añadido en la página login.html el contenedor donde se mostrará FirebaseUI.
- Creación del archivo javascript *firebase.js* con las configuraciones necesarias para FirebaseUI.
 Incluida también las funciones para enviar el idToken generado al servidor.
- Añadido *firebase.js* en el footer de la página *login.html*.
- Añadidas las librerías firebase-app.js y firebase-auth.js, al igual que sus configuraciones en el footer de la web.
- Capturado el evento "onsubmit" del formulario cierre de sesión para así cerrar la sesión en Firebase si se hubiera usado previamente.

4.4.2. HU 10. Lista de favoritos

En esta historia de usuario se crea la funcionalidad de poder añadir películas a una lista de favoritos, para ello el usuario debe de haber iniciado sesión previamente. El proceso de añadir o eliminar una película de los favoritos se ha desarrollado usando AJAX para una mejor experiencia de usuario y el cual consiste en pulsar el icono con forma de corazón que aparece en cada película, dicho proceso tiene los siguientes tres estados:

- Icono de corazón de color blanco Figura 4.39(a), el cual quiere decir que la película no está en la lista de películas favoritas del usuario.
- Icono con forma de cirulo girando continuamente Figura 4.39(b), el cual quiere decir que el usuario ha pulsado el icono del corazón y está en proceso de añadir o eliminar la película de su lista de favoritos.
- Icono de corazón de color rojo Figura 4.39(c), el cual quiere decir que la película se encuentra en la lista de películas favoritas del usuario.



Figura 4.39: Películas favoritas

LISTADO DE TAREAS DE LA HISTORIA DE USUARIO:

- Creación de los siguientes elementos:
 - Clase FavoriteIdentity, la cual será un identificador compuesto por el id de la entidad Movie y el email de la entidad User.
 - Entidad **Favorite**, la cual estará compuesta por el identificador FavoriteIdentity para almacenar en la base de datos las películas favoritas de los usuarios.
 - Interfaces **IFavoriteDao** e **IFavoriteService**, las cuales tendrás los métodos para obtener las películas favoritas de un usuario al igual el total de ellas.
 - Clase **FavoriteServiceImpl**, que implementará los métodos de la interfaz IFavoriteService
- Controlador **FavoriteController** con las siguientes URLs:
 - base-url/favorite/add-movie, para añadir la película al listado de películas favoritas del usuario.
 - *base-url/favorite/remove-movie*, para eliminar la película del listado de películas favoritas del usuario.
- Añadido en el archivo *ajax.js* las funciones de añadir y eliminar una película de los favoritos.
- Modificación de la lógica de la vista de las películas donde aparece el icono de favoritos, para que cuando dicha película sea favorita del usuario, el icono esté de color rojo y en caso contrario de color blanco.

4.4.3. HU 11. Detalle de lista de favoritos

La historia de usuario consiste en darle la opción al usuario de poder ver sus películas favoritas, para ello se ha incluido un icono en forma de corazón en el header de la web, el cual muestra el número de películas favoritas que tiene el usuario y al pulsarlo se le redirigirá a otra página con el detalle de su lista de películas favoritas.

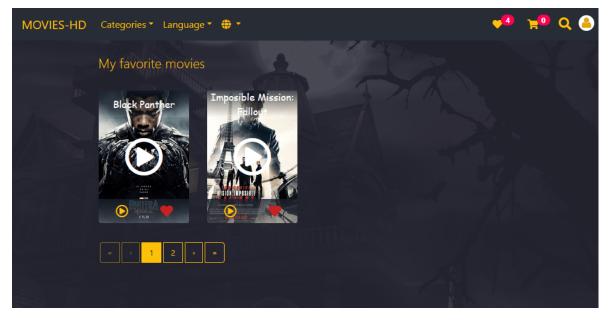


Figura 4.40: Listado de películas favoritas

LISTADO DE TAREAS DE LA HISTORIA DE USUARIO:

 Añadido en FavoriteController la URL base-url/favorite/all para obtener el listado de las películas favoritas de un usuario.

- Añadido el icono de favoritos con el enlace para obtener las películas favoritas en el header de la web, el cual se mostrará solo cuando el usuario haya iniciado sesión previamente.
- Creación de la página favorite_movies.html para mostrar el listado de las películas favoritas del usuario.

4.4.4. HU 16. Editar datos personales

En esta historia de usuario se ha creado una página de perfil, la cual está dividida en dos secciones, la primera es donde se muestra la imagen de perfil cuando se inicia sesión con Facebook o Google, aparte del nombre completo del usuario, el número de películas que tiene comprada y el número de películas que tiene en su lista de favoritos. La otra sección contiene varios formularios donde, aparte de mostrar todos los datos del usuario, también le permite editarlos.

Cada formulario tiene un botón que le permite habilitar la opción de modificar sus datos, una vez que se haga cualquier tipo de cambio en los datos se le mostrará el botón para guardar dichos cambios.

Listado de tareas de la historia de usuario:

- Creación de la entidad ContactInformation, para almacenar los datos de contacto del usuario.
- Añadida en la entidad User la relación @OneToOne con la entidad ContactInformation.
- Añadido en UserController las siguientes URLs:
 - base-url/profile, para mostrar la página del perfil del usuario.
 - base-url/user/profile, para editar los datos personales y de contacto del usuario.
 - base-url/user/password, para cambiar la contraseña de acceso del usuario.
- Creación de la página *profile.html* para mostrar todos los datos del usuario, el número de películas compradas, al igual que sus favoritas.
- Añadidas las reglas de estilo en el archivo styles.css para el diseño de la página creada en el punto anterior.
- Incluido en el menú de opciones del header de la web el enlace para ver el perfil de usuario.

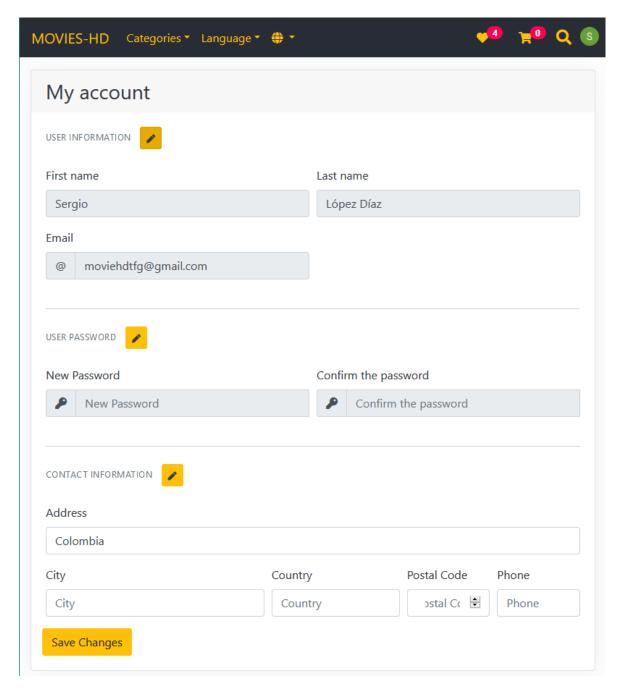


Figura 4.41: Formularios para editar los datos personales y de acceso

4.4.5. HU 8. Paginación

Esta historia de usuario consiste en la creación de una paginación para mostrar una cierta cantidad de películas por número de página, mejorando así la experiencia de usuario y la optimización en el proceso de recuperación de las películas desde la base de datos al no mostrarlas todas en la misma página de una sola vez. La paginación tiene las opciones de adelantar y retroceder una página, al igual que la opción de volver a la primera o última página.

LISTADO DE TAREAS DE LA HISTORIA DE USUARIO:

- Creación de los siguientes elementos:
 - Módulo paginator para estructurar las clases necesarias para el desarrollo de la paginación.

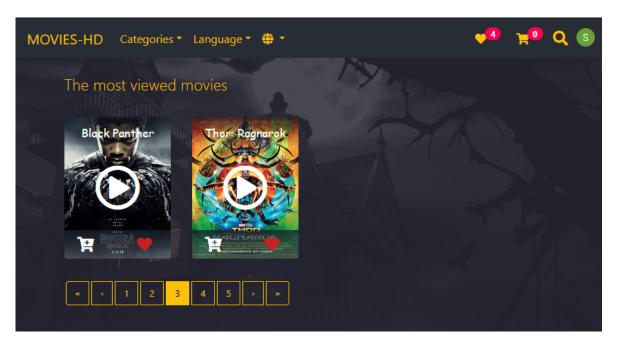


Figura 4.42: Paginación

- Clase **PageItem** que almacenará el número de página actual, al igual que un variable booleana para saber si es la página actual o no.
- Clase PageRender, que será la encargada de generar las diferentes URLs para la paginación.
- Clase **PaginationFactory** para estructurar el número de películas por página y el número de páginas a mostrar en la paginación.
- Fragmento *paginator_nav.html* que no es más que una plantilla con la estructura de la paginación a mostrar en las diferentes secciones de la web donde se necesite.
- Añadidas las reglas de estilos en el archivo s*tyles.css* para el diseño de la plantilla de paginación creada en el punto anterior.
- Refactorización de las interfaces IMovieDao e IMovieService y el servicio MovieServiceImpl cambiando algunos de sus métodos para manejar la paginación con las clases Pageable y Page.
- Actualización de MovieController para hacer uso de los nuevos métodos y clases para la paginación.
- Refactorización de las interfaces IUserDao e IUserService y el servicio UserServiceImpl cambiando algunos de sus métodos para manejar la paginación con las clases Pageable y Page.
- Actualización de UserController para hacer uso de los nuevos métodos y clases para la paginación.
- Refactorización de las interfaces IFavoriteDao e IFavoriteService y el servicio FavoriteService-Impl cambiando algunos de sus métodos para manejar la paginación con las clases Pageable y Page.
- Actualización de FavoriteController para hacer uso de los nuevos métodos y clases para la paginación.
- Añadida la plantilla *paginator_nav.html* en las diferentes secciones requeridas de la web para mostrar la paginación.

4.4.6. HU 21. Preguntas frecuentes

En esta historia de usuario se ha creado la página de preguntas frecuentes, las cuales están estructuradas en las siguientes secciones:

- **Plataforma**, que es donde se muestran los dispositivos de donde es posible acceder a MoviesHD.
- Contenido, donde se habla de los proveedores de contenidos cinematográfico.
- **Técnico**, donde se mencionan algunos de los problemas técnicos que se pueden tener a la hora de acceder a MoviesHD.
- Facturación y Términos, donde se mencionan los métodos de pago que existen, donde poder ver el historial de pagos y los términos y condiciones de uso de la web.



Figura 4.43: Pie de página de la web

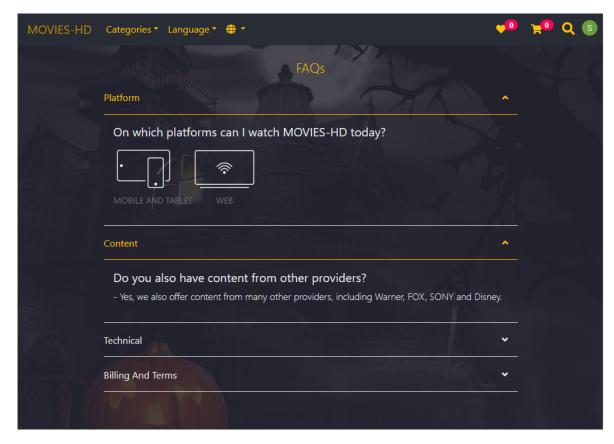


Figura 4.44: Página de preguntas frecuentes

Listado de tareas de la historia de usuario:

• Añadido al footer de la web la sección de preguntas frecuentes.

• Añadidas las reglas de estilos en el archivo styles.css para la sección creada en el punto anterior.

- Creación del controlador InfoController con la URL base-url/info/faqs para mostrar las preguntas frecuentes.
- Añadida la URL creada en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario acceder a la página de preguntas frecuentes.
- Creación de la página *faqs.html* para mostrar las preguntas frecuentes.
- Añadidas las reglas de estilos en el archivo styles.css para el diseño de la página de preguntas frecuentes.
- Añadidas en los recursos web del proyecto las imágenes que se muestran en la página de preguntas frecuentes.

4.4.7. HU 22. Contacto

La historia de usuario consiste en la creación de una página de contacto, en la cual se muestran ciertos datos de contacto como, por ejemplo, el correo electrónico, el número de teléfono y los horarios de la semana para poder contactar con MoviesHD. Por otro lado, se muestra el mapa de Google con la ubicación de MoviesHD.

LISTADO DE TAREAS DE LA HISTORIA DE USUARIO:

- Añadido al footer de la web la sección de contacto.
- Añadidas las reglas de estilos en el archivo styles.css para la sección creada en el punto anterior.
- Añadido en InfoController la URL *base-url/info/contact* para mostrar la página de contacto.
- Añadida la URL creada en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario acceder a la página de contacto.
- Creación en la consola de Google Cloud el API KEY para hacer uso de Google Maps[40].
- Creación de la página contact.html para mostrar la página de contacto y la ubicación de MoviesHD con Google Maps.
- Añadidas las reglas de estilos en el archivo styles.css para el diseño de la página de contacto.

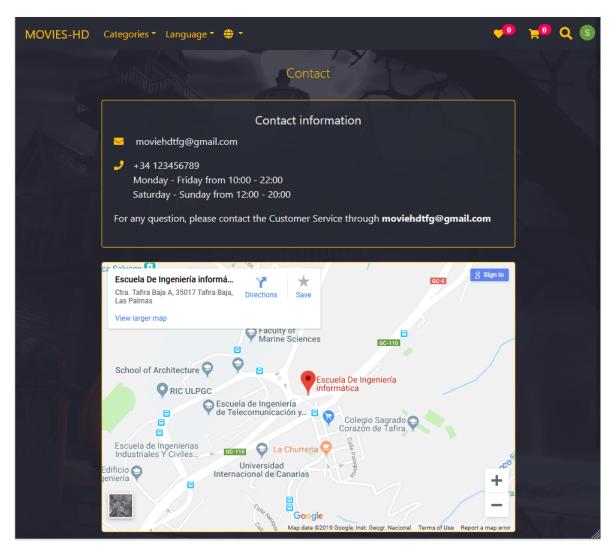


Figura 4.45: Página de contacto

4.4.8. HU 23. Política de datos personales

En esta historia de usuario se ha creado una página que contiene la política de datos personales. El principio de la página contiene un resumen del tratamiento que reciben los datos personales de los usuarios. A continuación de eso se encuentra una descripción más detallada del uso que se les da a los datos personales.

LISTADO DE TAREAS DE LA HISTORIA DE USUARIO:

- Añadido al footer de la web la sección de política de datos personales.
- Añadidas las reglas de estilos en el archivo *styles.css* para el diseño de la sección creada en el punto anterior.
- Añadido en InfoController la URL base-url/info/policy para mostrar la política de datos personales.
- Añadida la URL creada en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario acceder a la página políticas de datos personales.
- Creación de la página *data_policy.html* para mostrar la política de datos personales.

 Añadidas las reglas de estilos en el archivo styles.css para el diseño de la página de política de datos personales.

4.4.9. HU 24. Sobre nosotros

En esta historia de usuario se ha creado una página donde se habla de MoviesHD. El contenido consiste principalmente en la descripción del Trabajo de Fin de Grado.

Listado de tareas de la historia de usuario:

- Añadido al footer de la web la sección sobre nosotros.
- Añadidas las reglas de estilos en el archivo styles.css para el diseño de la sección creada en el punto anterior.
- Añadido en InfoController la URL *base-url/info/about* para mostrar la sección sobre nosotros.
- Añadida la URL creada en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario acceder a la página que describe a MoviesHD.
- Creación de la página *about.html* para mostrar la sección sobre nosotros.
- Añadidas las reglas de estilos en el archivo styles.css para el diseño de la página sobre nosotros.

4.4.10. HU 25. Términos y condiciones

En esta historia de usuario se ha creado una página web que contiene los términos y condiciones relacionados con el uso de MoviesHD.

Listado de tareas de la historia de usuario:

- Añadido al footer de la web la sección de términos y condiciones.
- Añadidas las reglas de estilos en el archivo styles.css para la sección creada en el punto anterior.
- Añadido en InfoController la URL base-url/info/terms para mostrar los términos y condiciones.
- Añadida la URL creada en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario acceder a la página de términos y condiciones de uso.
- Creación de la página terms_and_conditions.html para mostrar los términos y condiciones.
- Añadidas las reglas de estilos en el archivo styles.css para el diseño de la página de términos y condiciones.

4.4.11. HU 20. Multi-idioma

En esta historia de usuario se ha creado la funcionalidad de poder cambiar el idioma de la web. Los idiomas que dispone actualmente MoviesHD son español, inglés, alemán y francés. La implementación de esta funcionalidad no es compleja, solo que ha llevado bastante tiempo la creación de la misma, debido principalmente a la traducción de todo el texto de las páginas de la web.

Para la traducción de los idiomas se ha usado el traductor de Google. Los textos con los diferentes idiomas se guardan en archivos dentro de los recursos web del proyecto con la estructura

messages_locale.properties, donde se sustituye locale por el código de dos caracteres del idioma, por ejemplo, *messages_es.properties* para el idioma español.

Por otro lado, se ha tenido que cambiar los idiomas del sistema de inicio de sesión realizado con FirebaseUI, la pasarela de pago realizada con el Drop-in UI de Braintree, el reproductor de películas mediante Plyr y el reCaptcha de Google añadido en los formularios de registro y recuperación de la contraseña.

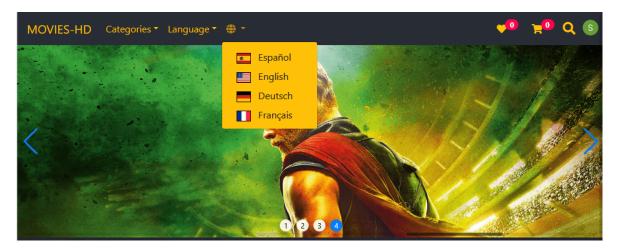


Figura 4.46: Menú de opciones de idioma de la web

LISTADO DE TAREAS DE LA HISTORIA DE USUARIO:

- Añadidos los beans LocaleResolver y LocaleChangeInterceptor en la clase de configuración MvcConfig para el cambio de idioma en la web.
- Creación del controlador LocaleController con la URL base-url/locale para gestionar el cambio de idioma.
- Añadida la URL creada en el punto anterior a los archivos de configuración de seguridad DevSpringSecurityConfig y ProductionSpringSecurityConfig y así permitirle a cualquier usuario poder cambiar el idioma de la web.
- Creación de los siguientes archivos:
 - messages_es.properties para almacenar las variables con el idioma en español.
 - messages_en.properties para almacenar las variables con el idioma en inglés.
 - messages de.properties para almacenar las variables con el idioma en alemán.
 - messages_fr.properties para almacenar las variables con el idioma en francés.
- Creación del módulo info dentro de los recursos web del proyecto para almacenar los siguientes módulos por idiomas:
 - Módulo **es**, para almacenar las páginas de información en el idioma español.
 - Módulo **en**, para almacenar las páginas de información en el idioma inglés.
 - Módulo de, para almacenar las páginas de información en el idioma alemán.
 - Módulo **fr**, para almacenar las páginas de información en el idioma francés.
- Actualización de los siguientes elementos:
 - Todas las páginas de la web para hacer uso de las variables creadas en los archivos messages de cada idioma.

 Todas las plantillas de correos electrónicos para hacer uso de las variables creadas en los archivos messages de cada idioma.

- Servicio MailClientBuilder añadiendo la clase Locale para el cambio de idioma en los correos electrónicos enviados.
- Todos los controladores para hacer uso de las variables creadas en los archivos messages de cada idioma.
- El componente LoginSuccessHandler para hacer uso de las variables creadas en los archivos messages de cada idioma.
- El componente InvoicePdfView para hacer de las variables creadas en los archivos messages de cada idioma y así modificar el idioma de las facturas creadas.
- Sistema de inicio de sesión de Google o Facebook añadiéndole la funcionalidad del multiidioma.
- El reCaptcha de Google añadiéndole la funcionalidad del multi-idioma.
- El sistema de pago añadiéndole la funcionalidad del multi-idioma al Drop-in UI de Braintree.
- Reproductor de vídeo Plyr añadiéndole la funcionalidad del multi-idioma.

4.4.12. Retrospectiva del Sprint

Con la implementación de este sprint el usuario puede iniciar sesión en MoviesHD mediante Facebook o Google y cambiar el idioma de la web a español, inglés, francés y alemán. El usuario también tiene la opción de crear una lista de películas favoritas para tener acceso a ellas en cualquier momento y poder comprarlas o ver sus detalles.

Por otra parte, el usuario puede conocer un poco más sobre MoviesHD, saber el trato que se le dan a sus datos personales, conocer los términos y condiciones de uso de la web, consultar las preguntas frecuentes sobre el uso de MoviesHD, etc.

En cuanto a las historias de usuarios a destacar en este Sprint se encuentran: el sistema de inicio de sesión con proveedores de servicios como Facebook o Google, permitiéndole al usuario crear e iniciar sesión en la web de manera más rápida y cómoda. Por otro lado, la historia de usuario del multi-idioma, permitiendo así que personas de otros países que hablan otro idioma puedan hacer uso y disfrute de MoviesHD.

No tanto dificultades, pero si arduos trabajos han requerido las historias de usuarios de la paginación y la de multi-idioma. Para la paginación se ha tenido que cambiar todos los métodos de donde se obtienen las películas para hacer uso de nuevos elementos que permiten crear dicha paginación. En cuanto a la funcionalidad del multi-idioma, es donde más tiempo se ha tenido que dedicar, por una parte, por la creación de los cuatro archivos messages donde se han almacenados los cuatro idiomas usados en MoviesHD y por otra parte porque se ha tenido que cambiar gran parte del código Java para el uso de las variables almacenadas en los archivos messages, al igual que en todas las páginas web de la aplicación.

4.4.13. Mejoras del Sprint

- Añadida la información de contacto del usuario en la factura de la compra.
- Añadida la funcionalidad de mostrar la imagen del proveedor de servicio tanto en el menú de opciones del header de la web, como en el perfil de usuario, todo eso si se inicia sesión con Google o Facebook.
- Añadido el *favicon.ico* de la web para que se muestre en la barra de navegación.

CAPÍTULO 5

CONCLUSIONES Y TRABAJO FUTURO

5.1. CONCLUSIONES

El desarrollo de MoviesHD se ha llevado a cabo con buenos resultados y los objetivos inicialmente planteados se han alcanzado satisfactoriamente.

El uso de SCRUM como metodología de desarrollo ágil ha tenido una gran importancia en el proceso de creación de la aplicación. Su uso ha permitido estructurar y llevar a cabo todas las fases del proyecto, priorizarlas para una buena complementación entre ellas y evitar circunstancias no deseadas.

En cuanto a las herramientas y tecnologías utilizadas caben destacar las siguientes:

- Los frameworks de Spring, ya que son la base del proyecto y sin ellos no se podría haber realizado. Con estos frameworks se desarrolla toda la lógica del servidor y el poder gestionar las peticiones y acciones de los usuarios.
- Bootstrap, con el cual se ha diseñado la mayor parte de la interfaz de usuario de la aplicación web.
- Amazon Web Services, debido a la variedad de servicios usados para poder almacenar los vídeos, imágenes y subtítulos de las películas. La base de datos donde almacenar toda la información relacionada con la web. El servicio de DNS para su uso en la creación de un dominio público y así se pueda tener acceso a la web desde cualquier sitio del mundo con acceso a Internet.
- La API de Braintree para la creación y gestión de la plataforma de pago, debido a que es una de las funcionalidades más importante del proyecto.
- El reproductor multimedia Plyr que es otro de los aspectos importantes en la aplicación, permitiendo al usuario ver las películas compradas.

Los resultados planteados en la introducción de este documento se han alcanzado con creces. MoviesHD tiene una interfaz de usuario amigable, fácil de usar y bastante atractiva de cara a los usuarios. Todas las funcionalidades implementadas tienen un objetivo claro y necesario de la idea del proyecto. La pasarela de pago le permite al usuario realizar la comprar sin ningún tipo de problema y mediante diferentes métodos de pago. Los usuarios pueden visualizar las películas compradas en cualquier momento del día y mediante dispositivos móviles, tabletas u ordenadores. El acceso a MoviesHD se puede hacer a través del dominio público www.movies-hd.cf, el cual hace uso de un certificado digital para cifrar todo el tráfico de la web y así proteger toda la información que circula a través de ella.

5.2. TRABAJO FUTURO

A continuación, se expone una serie de mejoras y funcionalidades que se podrían implementar en un futuro:

- Creación de una sección de administración donde se puedan controlar varios aspectos de la web, como pueden ser, la puesta de una película a la venta, lo cual consistiría en crear la película en Amazon RDS y en subir a Amazon S3 el archivo de vídeo, las imágenes de miniaturas para el reproductor de vídeo, las imágenes de poster y portada y los diferentes subtítulos de la película. Ponerle precios a las películas, añadirles los géneros de las mismas, los autores, el idioma, etc.
- Implementar el alquiler de las películas, y así ser más competitivo con la situación actual del mercado.
- Añadir opciones de suscripciones para adaptarse más al mercado actual.
- Mostrar las calificaciones de los usuarios de páginas como IMDB y The Movie DB para cada película.
- Creación de una aplicación móvil para los sistemas operativos iOS y Android.
- Realización de todos los trámites legales para que MoviesHD sea un comercio electrónico real.
- Permitirle al usuario cambiar la imagen de perfil.

NORMATIVA Y LEGISLACIÓN

A.1. LEYES DE PROTECCIÓN DE DATOS

El Portal Web MoviesHD al trabajar con datos personales debe de cumplir con una series de requisitos legales. El REGLAMENTO (UE) 2016/679 que se detallará más adelante obliga a personas, empresas y organismos de los Estados Miembros de la Unión Europea, tanto públicos como privados que manejen datos de carácter personal, a cumplir y aplicar una serie de medidas de seguridad para la protección de los datos que posean.

En MoviesHD se llevan a cabo dichas medidas de seguridad. El enlace https://www.movies-hd.cf/info/policy contiene en detalle el tratamiento que reciben los datos personales de los usuarios.

REGLAMENTO (UE) 2016/679, **2016 DEL PARLAMENTO EUROPEO Y DEL CONSEJO de 27 de abril de 2016**, relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la Directiva 95/46/CE (Reglamento general de protección de datos)[41].

Resumen de los rasgos fundamentales del reglamento EU

1. La figura del Delegado de Protección de Datos. El Reglamento impone la creación de una nueva figura llamada **Delegado de Protección de Datos (DPO, en inglés)**.

El delegado de protección de datos tendrá como mínimo las siguientes funciones:

- **Informar** y **asesorar** al responsable o al encargado del tratamiento y a los empleados que se ocupen del tratamiento de las obligaciones a las que se encuentran sujetos.
- Supervisar el cumplimiento del Reglamento, de otras disposiciones de protección de datos de la Unión o de los Estados miembros y de las políticas en materia de protección de datos personales, incluida la asignación de responsabilidades, la concienciación y formación del personal que participa en las operaciones de tratamiento, y las auditorías correspondientes.
- Asesoramiento acerca de la evaluación de impacto relativa a la protección de datos y supervisar su aplicación.
- Cooperar con la autoridad de control, actuando como interlocutor para cuestiones relativas al tratamiento.
- 2. Códigos de conducta y certificación o sellos de marca

Se va a promover la creación de mecanismos de certificación en materia de protección de datos y de sellos y marcas de protección de datos.

La certificación será voluntaria y deberá tener en cuenta las características específicas de los distintos sectores y las necesidades específicas de las microempresas y las pequeñas y medianas empresas.

Los organismos de certificación que tengan un nivel adecuado de pericia en materia de protección de datos expedirán y renovarán las certificaciones una vez informada la autoridad de control.

3. Consentimiento afirmativo para el tratamiento

El nuevo reglamento de protección de datos señala que todo tratamiento de datos personales debe ser lícito y leal.

El consentimiento para el tratamiento de los datos personales debe darse mediante un acto afirmativo claro que refleje una manifestación de voluntad libre, específica, informada, e inequívoca del interesado de aceptar el tratamiento de datos de carácter personal (declaración por escrito, una declaración verbal, marcar una casilla, escoger parámetros técnicos para la utilización de servicios informáticos...)

El silencio, las casillas ya marcadas o la inacción no constituye consentimiento.

El consentimiento debe darse para todas las actividades de tratamiento realizadas con el mismo o los mismos fines. Cuando el tratamiento tenga varios fines, debe darse el consentimiento para todos ellos.

4. Ejercicio de los derechos

Se hace especial hincapié en el principio de transparencia que exige que toda información y comunicación relativa al tratamiento de los datos personales sea fácilmente accesible y fácil de entender, en resumidas cuentas que se utilice un lenguaje sencillo y claro.

Como es hasta ahora deben quedar totalmente claros los siguientes puntos a la hora de solicitar o consultar información:

- Que se están recogiendo, utilizando, consultando o tratando datos personales.
- La medida en que dichos datos son o serán tratados
- La identidad del responsable del tratamiento
- Conocer los fines para los que se tratan los datos personales
- El plazo de tratamiento
- Los destinatarios si fuesen a cederse
- Los derechos que les asisten de ejercitar el derecho a presentar una reclamación ante una autoridad de control

5. Derecho al olvido

Los interesados deben tener derecho a que se rectifiquen los datos personales que le conciernen y un "derecho al olvido".

En el entorno en online, el derecho de al olvido incluye que quien publica datos personales está obligado a indicar a los responsables del tratamiento que estén tratando tales datos personales (sus fuentes) que supriman todo enlace a ellos , o las copias o réplicas de tales datos. Para entendernos, si ejerces tu derecho al olvido contra Google debe ser Google quien indique a las páginas web (periódicos, foros, redes, etc..) que supriman tus datos.

6. Limitación plazo de conservación de los datos personales

En relación plazo de conservación de los datos personales se debe garantizar que se limiten a un mínimo estricto, el responsable del tratamiento ha de establecer plazos para su supresión o revisión periódica. 7. Notificación de brechas de seguridad y exposición de los datos

Esto es una novedad destacable. El responsable de los datos debe notificar antes de 72 horas la violación de la seguridad de los datos personales a la autoridad de control competente y, en cooperación con la autoridad de control o policiales, notificar el riesgo al interesado permitiéndole tomar las precauciones necesarias.

Si dicha notificación no es posible en el plazo de 72 horas, debe acompañarse de una indicación de los motivos

8. No será necesario la inscripción de ficheros en la Agencia de Protección de Datos

Otra novedad importante es la eliminación de la obligación del registro de ficheros en la Agencia Española Protección de Datos (AEPD).

Se deben sustituir por procedimientos y mecanismos que se centren, en los tratamientos con un alto riesgo para los derechos y libertades de las personas físicas debido al tipo de datos o las formas de tratarlos.

Los encargados o el representante del encargado, debe mantener un registro de todas las actividades y categorías de actividades de tratamiento.

Estas obligaciones no se aplicarán a ninguna empresa ni organización que emplee a menos de 250 personas, a menos que el tratamiento que realice pueda entrañar un riesgo.

A.2. LEYES DE DERECHO DE AUTOR

En el Portal Web MoviesHD, los vídeos de las películas mostradas provienen de la plataforma de Youtube. No obstante, se debe conocer las leyes de derechos de autor relacionadas al contenido audiovisual para la posible implementación de uno de los trabajos futuros del proyecto, como es el de realizar todos los trámites legales para que MoviesHD sea un comercio electrónico real. A continuación se resume la Directiva 2001/29/CE del Parlamento Europeo y del Consejo respecto a los derechos de autor.

DIRECTIVA 2001/29/CE DEL PARLAMENTO EUROPEO Y DEL CONSEJO, DE 22 DE MAYO DE 2001, relativa a la armonización de determinados aspectos de los derechos de autor y derechos afines a los derechos de autor en la sociedad de la información[42].

Los derechos de autor son de dos clases:

- Derechos morales, irrenunciables e inalienables, como el derecho de reconocimiento de autoría y el derecho de integridad de la obra, entre otros
- Derechos económicos, transferibles y de duración limitada en el tiempo, básicamente los derechos de explotación, aunque hay otros (i.e. derecho a remuneración por copia privada)

Se considera autor a la persona natural que aparece como tal en la obra. En algunos casos previstos por la ley las personas jurídicas pueden tener algunos derechos económicos de propiedad intelectual.

En las obras en colaboración los derechos pertenecen a todos los autores; sin embargo, en las obras colectivas corresponden a la persona bajo cuya iniciativa y coordinación se edita y divulga la obra.

Todas las creaciones originales literarias, artísticas o científicas expresadas por cualquier medio o soporte, tangible o intangible, actualmente conocido o que se invente en el futuro.

La ley establece como derechos de explotación los siguientes:

 Reproducción: acto de fijación de la obra en un medio que permita su comunicación y la obtención de copias de todo o parte de ella.

- Distribución: acto de puesta a disposición del público del original o copias de la obra mediante su venta, alquiler, préstamo o cualquier otra forma.
- Comunicación pública: acto por el cual una pluralidad de personas pueda tener acceso a la obra sin previa distribución de ejemplares a cada una de ellas.
- Transformación: acto de traducción, adaptación y cualquier otra modificación de una obra en su forma de la que se derive una obra diferente. En el caso de las bases de datos, se considera transformación su reordenación.

Los derechos de explotación corresponden exclusivamente al autor y no pueden ser realizados sin su autorización, salvo en los casos previstos en la ley. El autor puede ceder todos o algunos de los derechos de explotación, según el tipo de contrato que firme.

Excepciones de los derechos de autor, también llamados límites, son casos en los que la ley autoriza el ejercicio de actos de explotación sin necesidad de una autorización por parte del titular de los derechos.

No es obligatorio inscribir las obras en el Registro de la Propiedad Intelectual. Los derechos de autor nacen en el momento de la creación de la obra. El Registro es sólo un medio de protección y de prueba de los derechos.

El copyright es la formula anglosajona para designar únicamente los derechos de explotación de una obra, sin referencia a los derechos morales. El símbolo © asociado a un nombre indica titularidad de derechos de explotación. Normalmente, aunque no necesariamente, va seguido de la expresión "todos los derechos reservados"

BIBLIOGRAFÍA

- [1] IntelliJ IDEA. JetBrain. url: https://www.jetbrains.com/idea (vid. pág. 25).
- [2] Sourcetree. Simplicity and power in a beautiful Git GUI. URL: https://www.sourcetreeapp.com (vid. pág. 25).
- [3] GIT. A distributed version control system designed to handle everything from small to very large projects with speed and efficiency. URL: https://git-scm.com (vid. pág. 25).
- [4] MySQL Workbench. A unified visual tool for database architects. URL: https://www.mysql.com/products/workbench (vid. pág. 25).
- [5] AWS. Amazon Web Services. URL: https://docs.aws.amazon.com (vid. pág. 25).
- [6] Amazon S3. Simple Storage Service. URL: https://aws.amazon.com/es/s3 (vid. pág. 25).
- [7] Amazon CloudFront. A fast content delivery network. URL: https://aws.amazon.com/es/s3 (vid. pág. 26).
- [8] Amazon RDS. Relational Database Service. URL: https://aws.amazon.com/es/rds (vid. pág. 26).
- [9] Amazon Elastic Beanstalk. *An easy-to-use service for deploying and scaling web applications and services.* URL: https://aws.amazon.com/elasticbeanstalk (vid. pág. 26).
- [10] Amazon Elastic Compute Cloud. A web service that provides secure, resizable compute capacity in the cloud. URL: https://aws.amazon.com/es/ec2 (vid. pág. 26).
- [11] Amazon Route 53. *A highly available and scalable cloud Domain Name System.* URL: https://aws.amazon.com/es/route53 (vid. pág. 26).
- [12] Amazon Certificate Manager. Easily provision, manage, and deploy public and private SSL/TLS certificates for use with AWS services. URL: https://aws.amazon.com/es/certificate-manager (vid. pág. 26).
- [13] IAM. Identity and Access Management. URL: https://aws.amazon.com/es/iam (vid. pág. 26).
- [14] Amazon VPC. Amazon Virtual Private Cloud. URL: https://aws.amazon.com/es/vpc (vid. pág. 26).
- [15] Spring Boot. Makes it easy to create stand-alone, production-grade Spring based Applications. URL: https://spring.io/projects/spring-boot (vid. pág. 26).
- [16] Spring Data. Makes it easy to use data access technologies, relational and non-relational databases. URL: https://spring.io/projects/spring-data (vid. pág. 26).
- [17] Spring Security. *A powerful and highly customizable authentication and access-control framework.* URL: https://spring.io/projects/spring-security (vid. pág. 26).
- [18] Spring Mobile. A framework that provides capabilities to detect the type of device making a request to your Spring web site. URL: https://projects.spring.io/spring-mobile (vid. pág. 26).
- [19] Thymeleaf. A modern server-side Java template engine for both web and standalone environments. URL: https://www.thymeleaf.org (vid. pág. 26).
- [20] Maven. A software project management and comprehension tool. URL: https://maven.apache.org (vid. pág. 26).

80 BIBLIOGRAFÍA

[21] Bootstrap. Build responsive, mobile-first projects on the web with the world's most popular front-end component library. URL: https://getbootstrap.com (vid. pág. 26).

- [22] jQuery. A fast, small, and feature-rich JavaScript library. URL: http://jquery.com (vid. pág. 26).
- [23] jQuery UI. A curated set of user interface interactions, effects, widgets, and themes built on top of the jQuery JavaScript Library. URL: https://jqueryui.com (vid. pág. 26).
- [24] Firebase Authentication. URL: https://firebase.google.com/docs/web/setup (vid. pág. 26).
- [25] JSON Web Token. Allows you to decode, verify and generate JWT. URL: https://jwt.io (vid. pág. 27).
- [26] Swiper. Most Modern Movile Touch Slider. URL: http://idangero.us/swiper (vid. pág. 27).
- [27] Plyr. A simple, accessible and customisable media player for Video, Audio, YouTube and Vimeo. URL: https://plyr.io (vid. pág. 27).
- [28] StarUML. A sophisticated software modeler for agile and concise modeling. URL: http://staruml.io (vid. pág. 27).
- [29] The Movie Database. *A community built movie and TV database*. URL: https://www.themoviedb.org (vid. pág. 29).
- [30] Fontawesome 5. Vector icons and social logos on your website with Font Awesome, the web's most popular icon set and toolkit. URL: https://fontawesome.com (vid. pág. 29).
- [31] Google reCaptcha. *It protects you against spam and other types of automated abuse.* URL: https://developers.google.com/recaptcha/intro (vid. pág. 38).
- [32] Freenom. A fast and anonymous Public DNS resolver. URL: https://my.freenom.com (vid. pág. 40).
- [33] PayPal Sandbox. A self-contained, virtual testing environment that mimics the live PayPal production environment. URL: https://developer.paypal.com/docs/classic/lifecycle/ug_sandbox (vid. pág. 43).
- [34] Braintree Drop-in UI. A complete, ready-made payment UI that offers a quick and easy way to securely accept payments. URL: https://developers.braintreepayments.com (vid. pág. 48).
- [35] Videohop. *It generates JPEG thumbnail sprite sheets for video seek preview.* URL: https://www.videohop.com.au/thumbnail-generator (vid. pág. 49).
- [36] JetS3t. A free, open-source Java toolkit and application suite for (Amazon S3), Amazon CloudFront, and Google Storage for Developers. URL: http://www.jets3t.org (vid. pág. 51).
- [37] OpenSSL. A robust and full-featured toolkit for the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols. URL: https://www.openssl.org (vid. pág. 52).
- [38] OpenPDF. A free Java library for creating and editing PDF files with a LGPL and MPL open source license. URL: https://github.com/LibrePDF/OpenPDF (vid. pág. 55).
- [39] Software Development Kit Firebase Admin. URL: https://firebase.google.com/docs/admin/setup (vid. pág. 60).
- [40] Google Maps. URL: https://console.cloud.google.com/google/maps-apis/overview (vid. pág. 68).
- [41] REGLAMENTO (UE) 2016/679. URL: https://eur-lex.europa.eu/legal-content/ES/TXT/PDF/?uri=CELEX:32016R0679&from=ES (vid. pág. 75).
- [42] DIRECTIVA 2001/29/CE DEL PARLAMENTO EUROPEO Y DEL CONSEJO. URL: https://eurlex.europa.eu/legal-content/ES/TXT/PDF/?uri=CELEX:32001L0029&from=ES (vid. pág. 77).