

CINEMAMI

Trabajo Final de Grado - Memoria

Alicia García Espiga

GRADO INGENIERÍA INFORMÁTICA
JAVAEE

Nombre Consultor/a: **Vicenç Font Sagrista**

Nombre Profesor/a responsable de la asignatura: **Santi Caballe Llobet**

Fecha Entrega: Enero de 2019

Agradecimientos

A mi marido, por recordarme cada día que podía con ello. Por estar con las niñas para que yo pudiera sacar tiempo.

A mis hijas por hacer que me distrajera en momentos agobiantes.

A mis padres, por apoyarme.

A mis amigas, por comprenderme.

A la UOC por hacer esto posible.

GRACIAS

C)Reconocimiento-NoComercial-SinObraDerivada



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada [3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>CINEMAMI</i>
Nombre del autor:	<i>Alicia García Espiga</i>
Nombre del consultor/a:	<i>Vicenç Font Sagrista</i>
Nombre del PRA:	<i>Santi Caballe Llobet</i>
Fecha de entrega (mm/aaaa):	01/2019
Titulación::	<i>GRADO INGENIERÍA INFORMÁTICA</i>
Área del Trabajo Final:	<i>JavaEE</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	<i>javaEE, Spring, MVC.</i>

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

Este documento consiste en la Memoria de la asignatura Trabajo Fin de Grado en el área JavaEE, donde se puede ver en detalle el proceso de creación de una aplicación Web utilizando como lenguaje de programación JAVA, con tecnologías y herramientas afines a este lenguaje.

Como madre de dos niñas impulsora de una nueva idea para la ayuda a las madres facilitando unas sesiones de Cine para poder acceder con sus bebés, nace la idea de la creación de una aplicación para ayudar a las administradoras a gestionar de modo centralizado las sesiones y los cines.

El proyecto CINEMAMI, consiste en una aplicación de gestión de sesiones de cine para bebés. Es un sistema privado y para acceder necesitará ser un usuario autenticado. Dependiendo del rol que tenga el usuario que acceda, podrá realizar unas funciones u otras.

La metodología que he utilizado es la aprendida durante la carrera de Ingeniería informática así como en mi experiencia profesional.

Como resultado tenemos una aplicación web para la gestión de sesiones de cine para ir con su bebé, que ha sido creada con mucho esfuerzo y dedicación. En la implementación he tenido que documentarme aprendiendo cada día para poder realizar una aplicación utilizando tecnologías de última generación.

Abstract (in English, 250 words or less):

This document is the thesis of the unit Final degree assignment in JavaEE area, where we can explore in detail the web application creation process using JAVA programming language, along with technology and tools related to this language.

As a mum of two children I was the driving force behind the initiative of offering baby- friendly cinema sessions.

CINEMAMI project is a management software that helps to organize and centralize these cinema sessions. Since it's a private system, users will need to be authenticated in order to access to it. The system will offer different functions depending on the user's profiles.

I have used different methodologies based on both, my previous professional experience and the knowledge acquired in my Computer Engineering degree.

As a result we have an application to manage cinema series aimed at mums and babies, created with a lot of effort and dedication. For its implementation, I have done an extensive research, which has allowed me to create an application using the latest technology.

Índice

1.	Introducción	1
1.1	Contexto y justificación del Trabajo	1
1.2	Objetivos del Trabajo.....	2
1.3	Enfoque y método seguido.....	3
1.4	Planificación del Trabajo	3
1.5	Breve resumen de productos obtenidos	4
1.6	Descripción de las tecnologías a utilizar en el desarrollo.	5
1.7	Breve descripción de los otros capítulos de la memoria.....	5
2.	Requerimientos, Análisis y Diseño.....	6
2.1.	Actores (stakeholders).....	6
2.2.	Funcionalidades.....	6
2.3.	Requerimientos.....	6
2.4.	Fichas de Casos de Uso:.....	7
2.5.	Diagrama de Casos de Uso.....	15
2.6.	Diagrama de clases	16
2.7.	Diseño relacional de la base de datos	17
2.8.	Diagrama de componentes.....	18
2.9.	Diagrama de arquitectura	20
2.10.	Arquitectura	37
2.10.1.	<i>Spring MVC</i>	37
2.10.2.	<i>Otras tecnologías</i>	38
2.11.	Análisis de Riesgos	38
2.12.	Prototipo gráfico inicial	40
3.	Implementación y programación	48
3.1.	Experiencia en el desarrollo del proyecto:	48
3.2.	Mejoras para siguientes versiones:	49
3.3.	Resultado role Admin:	50
3.4.	Resultado role USER:.....	56
4.	Pruebas	60
4.1.	Resumen de las pruebas realizadas por sistema:	60
4.2.	Detalle de las pruebas realizadas por componente y caso de uso:	60
5.	Conclusiones	65
6.	Bibliografía	66
7.	Anexos	67
7.1.	Manual de instalación:	67

Lista de figuras

Ilustración 1 - Resumen tareas Diagrama Gantt	4
Ilustración 2 - Diagrama Gantt	4
Ilustración 3 - Diagrama de Casos de Uso	15
Ilustración 4 - Diagrama de Clases	16
Ilustración 5 – Diseño relacional de la BBDD	17
Ilustración 6 – Diagrama de componentes	18
Ilustración 7 -Interfaz Componentes Capa Presentación	19
Ilustración 8 – Interfaz Componentes Capa Negocio	19
Ilustración 9 - Interfaz Componentes Capa Integración	20
Ilustración 10 - Refinamiento - Presentación	21
Ilustración 11 – Diseño Componente Evaluacion Refinado (Presentacion)	24
Ilustración 12 - Diseño Componente FilmAdmin Refinado (Presentacion)	25
<i>Ilustración 13 - Diseño Componente CinemaAdmin Refinado (Presentacion)</i>	26
Ilustración 14 - Diseño Componente SystemAdmin Refinado (Presentacion)	27
Ilustración 15 - Diseño Componente EvaluationAdminPresentation JavaEE	28
Ilustración 16 - Diseño Componente FilmAdminPresentation JavaEE	29
Ilustración 17 - Diseño Componente CinemaAdminPresentation JavaEE	30
Ilustración 18 - Diseño Componente SystemAdminPresentation JavaEE	31
Ilustración 19 - Diseño de los Componentes para la capa de Negocio.	32
Ilustración 20 - Diagrama Evaluacion capa Negocio JavaEE	33
Ilustración 21 – Diagrama película capa Negocio JavaEE	33
Ilustración 22 - Diagrama cine capa Negocio JavaEE	34
Ilustración 23 - Diagrama SystemAdmin capa de Negocio JavaEE	34
Ilustración 24 - Diagrama de componentes capa Integración	35
Ilustración 25 - Diagrama EvaluantionAdmin capa Integration JavaEE	35
Ilustración 26 - Diagrama FilmAdmin capa Integración JavaEE	36
Ilustración 27 - Diagrama CinemaAdmin capa Integración JavaEE	36
Ilustración 28 - Diagrama SystemAdmin capa Integración JavaEE	36
Ilustración 29 - Diagrama dependencias Integración - punto de vista info.	37
Ilustración 30 - Spring MVC	38
Ilustración 31 - Analisis de Riesgos	39
Ilustración 32 - Prototipo Acceso	40
Ilustración 33 - Prototipo Cines	41
Ilustración 34 - Prototipo Alta Cine	41
Ilustración 35 - Prototipo Detalle Cine	42
Ilustración 36 - Prototipo Actualizar Cine	42
Ilustración 37 - Prototipo Borrar Cine	43
Ilustración 38 - Prototipo Películas	43
Ilustración 39 - Prototipo Alta Película	44
Ilustración 40 - Prototipo Detalle Película	44
Ilustración 41 - Prototipo Actualizar Película	45
Ilustración 42 - Prototipo Eliminar Película	45
Ilustración 43 - Prototipo Valorar Película	46
Ilustración 44 - Prototipo Buscar Película	46
Ilustración 45 - Prototipo Ver todas las Evaluaciones	47
Ilustración 46 - Prototipo Ver evaluaciones y cometarios de una película	47

1. Introducción

1.1 Contexto y justificación del Trabajo

El punto inicial para la realización de mi Trabajo Final de Grado(a partir de ahora TFG), ha sido el repaso de cada una de las asignaturas cursadas durante la carrera.

Aunque inicialmente me planteé desarrollar una aplicación usando .Net, pensé que si lo hacía usando una tecnología la cual conozco bien, no me iba a aportar conocimientos adicionales y no iba a suponer un gran reto para mí. Así que pensé que podría ser un buen objetivo el reto de aprender a utilizar una tecnología de última generación como es Java EE.

A parte de ser un gran reto, sé que puede ser un problema por la falta de conocimientos que tengo de esta tecnología, pero la manera de aprender es ponerse con ello.

La motivación de realizar una aplicación para ayudar en la gestión de las sesiones de cine para los bebés, ha sido porque tengo dos niñas pequeñas y desde que nació la mayor, siempre he acudido a los talleres de lactancia. Allí, una enfermera de pediatría nos propuso que para desconectar podríamos ir un día al cine con nuestros bebés. Esta enfermera, propuso a unos cines hacer una sesión especial adaptada para los bebés tanto en sonido como en temperatura. Y de esta manera, las madres disfrutábamos de las películas de actualidad y sobre todo nos daba la oportunidad de hacer algo diferente. Como las sesiones fueron un éxito (tanto que a día de hoy siguen existiendo), yo me propuse el seguir con ello informando a través de Facebook (sesión teta al cine con tu bebé #sesionteta) de todas las sesiones que nos van diciendo los cines.

Con esta idea, se me ha ocurrido, que como ha ido creciendo tanto en número de cines que tienen este tipo de sesión como de colaboradores, que para facilitar la gestión de los cines y de las películas, podría hacer una aplicación utilizando tecnología Java EE.

La situación actual es que los cines se ponen en contacto con un grupo de madres y éstas me envían la información para colgar en Facebook. Está todo el trabajo centralizado en una única persona y al haber cada vez más cines y más películas que gestionar, se va haciendo cada día más complicada su gestión.

Con el desarrollo de esta aplicación web se pretende ayudar a mejorar la gestión, ayudando al grupo de madres a centralizar la información. Se podría mejorar para que las madres pudieran valorar las películas y

ayudar a elegir a qué cine ir según la valoración y los comentarios publicados.

1.2 Objetivos del Trabajo

Los objetivos de este trabajo, están realizados desde dos puntos de vista:

- Desde el punto de vista académico: tiene el objetivo de aplicar los conocimientos adquiridos durante toda la carrera. También tiene el objetivo de adquirir conocimientos de las nuevas tecnologías usando Java EE.
- Desde el punto de vista del software a desarrollar: debe poderse gestionar los cines con sus sesiones, automatizar el tráfico de información simplificando el procedimiento de envío, controlarse el acceso con diferentes perfiles, poder gestionar las películas y tener un mecanismo de evaluación de las mismas por perfiles.

Las funcionalidades principales serán:

- CU_1: Identificarse al sistema
- CU_2: Salir del sistema
- CU_3: Dar de alta un cine
- CU_4: Eliminar un cine
- CU_5: Modificar un cine
- CU_6: Ver detalle de un cine
- CU_7: Dar de alta una película
- CU_8: Eliminar una película
- CU_9: Modificar una película
- CU_10: Evaluar una película
- CU_11: Ver detalle de una película
- CU_12: Buscar una película aplicando filtros
- CU_13: Ver los comentarios y evaluación de una película
- CU_14: Eliminar comentarios y evaluación de una película
- CU_15: Ver todas las evaluaciones

Los requisitos funcionales pueden ser los siguientes:

- Para acceder a la aplicación los usuarios tendrán que identificarse en el sistema con un control de usuario y password.
- Habrá dos roles diferentes para acceder: uno de usuario y otro de administrador.
- Un usuario podrá ver los cines y las sesiones, valorar una película y buscar las películas.
- Un administrador tendrá acceso total al contenido del portal, pudiendo también valorar una película. El sentido de que el administrador pueda

valorar también una película es por los comentarios que pueda ver de alguna madre en las redes sociales.

- El administrador podrá además de lo que hace un usuario:
 - ❖ dar de alta y baja tanto cines como películas.
 - ❖ ver la valoración y comentarios de todas las películas.
 - ❖ buscar las valoraciones y comentarios de una película
- La aplicación debe ser desarrollada en JavaEE. El sentido de usar esta tecnología es porque no se quiere instalar software adicional en el puesto cliente y su acceso debe ser tipo web.

1.3 Enfoque y método seguido

Actualmente no existe nada similar en el mercado dada la especialidad del trabajo. Dado que el problema es que la gestión de estas sesiones tan particulares ha ido creciendo en el tiempo, ha ocasionado que el trabajo también haya ido creciendo al mismo ritmo.

El desarrollo de una aplicación web a medida es la solución que permitirá al grupo de madres interactuar con un sistema centralizado y ayudará a tener un control total. Con esta solución se podrá llegar a los objetivos marcados.

La metodología que voy a utilizar es la tradicional para la parte de diseño y análisis y entiendo que para el desarrollo podría utilizar una metodología *agile* para ir realizando pequeños desarrollos completos como piezas únicas para una implantación más rápida.

En la fase de diseño, utilizaré diagramas UML apoyándome en MagicDraw para dibujarlos.

1.4 Planificación del Trabajo

Hay que tener en cuenta que este TFG va a ser realizado por una sola persona, y que en caso de ser un proyecto para realizar a una empresa, habría que definir un equipo de trabajo con diferentes roles.

Para poder hacer una planificación del trabajo he realizado con Microsoft Project un diagrama de Gantt como se puede ver a continuación:

Resumen de Tareas Diagrama de Gantt:

ID	Task Mode	Task Name	Duration	Start	Finish
1		▶ PLAN DE TRABAJO (PEC1)	9 days	Thu 20/09/18	Mon 01/10/18
7		▶ REQUERIMIENTOS ANÁLISIS Y DISEÑO (PEC2)	26 days	Tue 02/10/18	Tue 06/11/18
19		▶ IMPLEMENTACIÓN (PEC3)	32 days	Wed 07/11/18	Thu 20/12/18
25		▶ MEMORIA Y PRESENTACIÓN	14 days	Fri 21/12/18	Wed 09/01/19
28		▶ TRIBUNAL	6 days	Mon 14/01/19	Mon 21/01/19

Ilustración 1 - Resumen tareas Diagrama Gantt

Diagrama de Gantt:

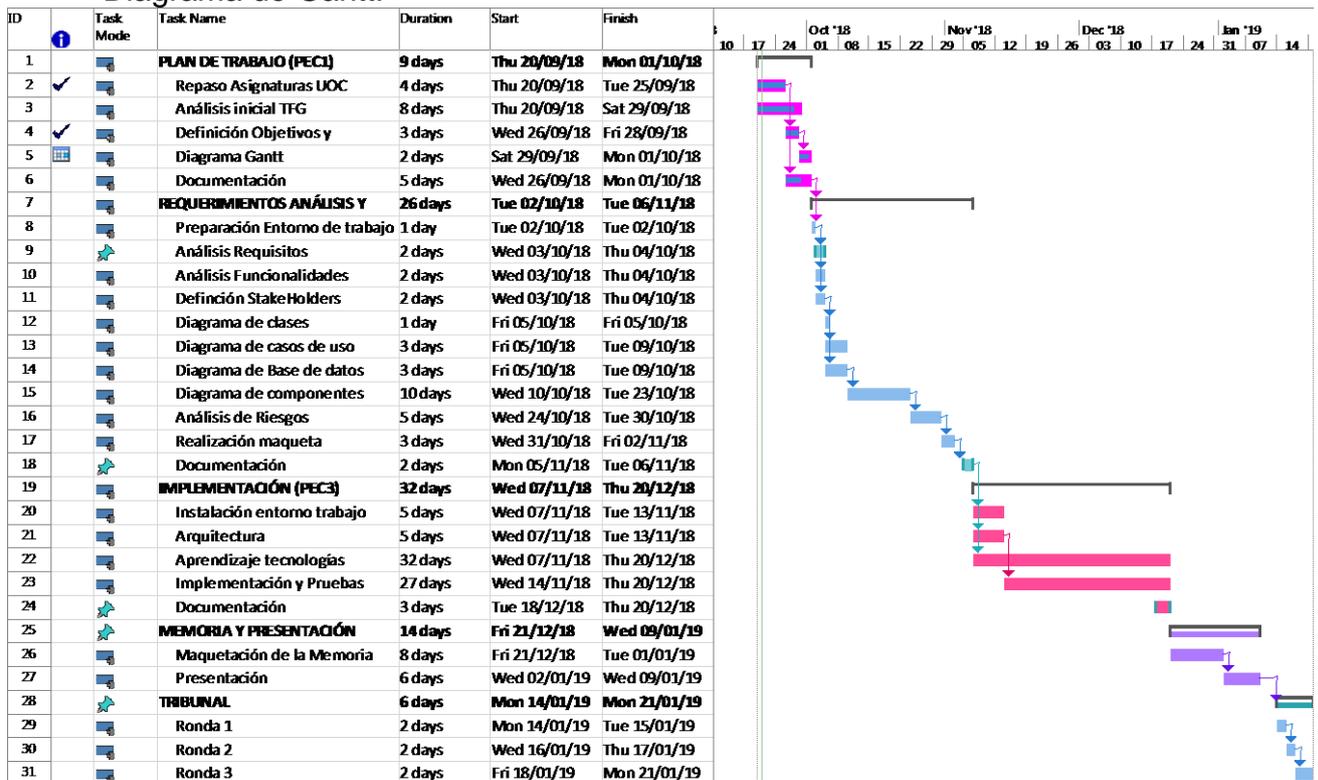


Ilustración 2 - Diagrama Gantt

1.5 Breve resumen de productos obtenidos

Los productos que se van a obtener serán los siguientes:

- Memoria y presentación
- Maqueta
- Software completo
- Manual de usuario

1.6 Descripción de las tecnologías a utilizar en el desarrollo.

El Framework que usaré será Spring porque según he estado investigando, es la tecnología que mejor se adapta al desarrollo que quiero realizar. Salvo que la experiencia en la fase de implementación sea contradictoria, utilizaré como servidor de aplicaciones embebido Spring-Boot. Parece que con Spring-Boot será más fácil la parte de configuración del entorno (en otros proyectos realizados en asignaturas que usaban JAVA, es la parte que más tiempo tuve que dedicar y por eso creo que puede ser una buena utilidad a la hora de implementar esta aplicación). Al utilizar este Framework, utilizaré JavaServer Pages (JSP) para la implementación de las vistas. Como gestor de base de datos utilizaré PostgreSQL.

1.7 Breve descripción de los otros capítulos de la memoria

En los siguientes capítulos voy a realizar el análisis y diseño, dónde realizaré un estudio detallado de los requisitos o especificaciones de la aplicación. Posteriormente, realizaré la implementación de la solución según el análisis y diseño anterior junto con las pruebas oportunas para dar por válidos los entregables según su estado. También incluiré unas conclusiones del trabajo realizado. Finalizaré la memoria introduciendo un glosario de términos y una bibliografía. Por último introduciré un apartado de anexos con los manuales y demás documentación a aportar.

2. Requerimientos, Análisis y Diseño

2.1. Actores (stakeholders)

La aplicación que se va a desarrollar va a tener los siguientes stakeholders; Administrador (A) y Usuario (U). Los Administradores son los encargados de administrar la aplicación, para gestionar los nuevos cines o películas y es el que va a tener control total sobre la misma. Los Usuarios son los que van a poder ver que películas hay en los cines que ofrecen la sesión especial para bebés y van a poder evaluar las películas para ayudar a otras madres a elegir la película que ver.

2.2. Funcionalidades

Las funcionalidades que debe la aplicación serán:

- CU_1: Identificarse al sistema (A, U)
- CU_2: Salir del sistema (A, U)
- CU_3: Dar de alta un cine (A)
- CU_4: Eliminar un cine(A)
- CU_5: Modificar un cine (A)
- CU_6: Ver detalle de un cine (A, U)
- CU_7: Dar de alta una película(A)
- CU_8: Eliminar una película (A)
- CU_9: Modificar una película (A)
- CU_10: Evaluar una película (A, U)
- CU_11: Ver detalle de una película (A, U)
- CU_12: Buscar una película aplicando filtros (A, U)
- CU_13: Ver evaluación de una película (A, U)
- CU_14: Ver los comentarios y la evaluación de una película (A)
- CU_15: Eliminar comentarios y evaluación de una película (A)
- CU_16: Ver todas las evaluaciones (A)

2.3. Requerimientos

Los requerimientos desde el punto de vista Funcional son:

- Para acceder a la aplicación los usuarios (Administradores y Usuarios) tendrán que identificarse en el sistema con un control de usuario y password.
- Habrá dos roles diferentes para acceder: uno de usuario y otro de administrador.
- Un usuario podrá ver los cines y las sesiones, valorar una película y buscar las películas.

- Un administrador tendrá acceso total al contenido del portal, pudiendo también valorar una película. El sentido de que el administrador pueda valorar también una película es para evaluarla por los comentarios que pueda ver de algunas madres en las redes sociales.
- El administrador podrá además de lo que hace un usuario:
 - ❖ dar de alta, baja y modificar tanto cines como películas.
 - ❖ ver la valoración y comentarios de todas las películas.
 - ❖ buscar las valoraciones y comentarios de una película

Desde el punto de vista de la tecnología:

- La aplicación debe ser desarrollada en JavaEE. El sentido de usar esta tecnología es porque no se quiere instalar software adicional en el puesto cliente y su acceso debe ser tipo web.

2.4. Fichas de Casos de Uso:

CU_1	LOGIN
Actor Principal	Administrador, Usuario
Objetivos	<ol style="list-style-type: none"> 1. Dar acceso al sistema 2. Asignar un rol
Pre-condiciones	No haberse logado.
Post-condiciones	El usuario se autentica en el sistema.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página para introducir el usuario y la contraseña. 2. El usuario introduce su usuario y su contraseña y pulsa el botón <i>Login</i>. 3. Se accede a la <i>Home</i>.
Escenario alternativo	<ol style="list-style-type: none"> 3. Las credenciales no son correctas. 4. El sistema muestra un mensaje de error. 5. El sistema vuelve a la pantalla de <i>Login</i>

CU_2	LOGOUT
Actor Principal	Administrador, Usuario
Objetivos	Salirse del sistema
Pre-condiciones	Haberse logado.
Post-condiciones	Se cierra la sesión del usuario. Se muestra la página de <i>Login</i>
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra un link con la opción del <i>Logout</i>. 2. El usuario hace clic en el link de <i>Logout</i>. 3. Se cierra la sesión del usuario 4. Se muestra la página de <i>Login</i>.

CU_3	DAR DE ALTA UN CINE
Actor Principal	Administrador
Objetivos	Dar de alta un nuevo cine en la Base de Datos
Pre-condiciones	Estar logado como Administrador El cine no existe en el sistema
Post-condiciones	El cine se graba en la Base de Datos.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con el botón Alta-Cine. 2. El usuario pulsa el botón Alta-Cine. 3. El sistema muestra una página con los datos de alta de cine. 4. El usuario rellena los datos solicitados. 5. El usuario pulsa el botón Guardar-Cine. 6. El sistema guarda el cine y muestra un mensaje de grabado satisfactoriamente.
Escenario alternativo	<ol style="list-style-type: none"> 6. Los datos introducidos no son válidos. 7. El sistema devuelve un error de validación. 8. El sistema vuelve a la pantalla de Alta-Cine

CU_4 ELIMINAR UN CINE	
Actor Principal	Administrador
Objetivos	Eliminar un cine en la Base de Datos
Pre-condiciones	Estar logado como Administrador El cine existe en el sistema
Post-condiciones	El cine se elimina de la Base de Datos.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con el botón Eliminar-Cine. 2. El usuario pulsa el botón Eliminar-Cine. 3. El sistema muestra una página con los datos del cine que queremos eliminar. 4. El usuario pulsa el botón Eliminar-Cine. 5. El sistema elimina el cine 6. El sistema muestra la lista de cines.
Escenario alternativo	<ol style="list-style-type: none"> 5. El sistema da un error de acceso a base de datos. 6. El sistema vuelve a la pantalla con los datos del cine a eliminar.

CU_5 MODIFICAR UN CINE	
Actor Principal	Administrador
Objetivos	Modificar un cine guardado en la Base de Datos
Pre-condiciones	Estar logado como Administrador El cine existe en el sistema
Post-condiciones	El cine se modifica en Base de Datos.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con el botón Modificar-Cine. 2. El usuario pulsa el botón Modificar-Cine. 3. El sistema muestra una página con los datos del cine que queremos Modificar 4. El usuario modifica los datos. 5. El usuario pulsa el botón Guardar. 6. El sistema guarda el cine 7. El sistema muestra la lista de cines.
Escenario alternativo	<ol style="list-style-type: none"> 6. El sistema da un error con los datos introducidos no son correctos. 7. El sistema vuelve a la pantalla con los datos del cine a modificar. 8. El sistema marca en rojo los datos erróneos.

CU_6 VER DETALLE DE UN CINE	
Actor Principal	Administrador, Usuario
Objetivos	Consultar el detalle de un cine guardado en la Base de Datos
Pre-condiciones	Estar logado en el sistema El cine existe en el sistema
Post-condiciones	Se muestra una página con el detalle del cine.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con el botón Ver-Detalle-Cine. 2. El usuario pulsa el botón Ver-Detalle-Cine. 3. El sistema muestra una página con los datos del cine que queremos Ver.
Escenario alternativo	<ol style="list-style-type: none"> 3. El sistema da un error de acceso a la base de datos. 4. El sistema vuelve a la pantalla para poder pulsar el botón Ver-Detalla-Cine.

CU_7 DAR DE ALTA UNA PELÍCULA	
Actor Principal	Administrador
Objetivos	Dar de alta una nueva película en Base de Datos
Pre-condiciones	Estar logado como Administrador La película no existe en el sistema
Post-condiciones	La película se graba en la Base de Datos.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con la relación de películas y con el botón Alta-Película. 2. El usuario pulsa el botón Alta-Película. 3. El sistema muestra una página con los datos de alta de la película. 4. El usuario rellena los datos solicitados. 5. El usuario pulsa el botón Guardar-Película. 6. El sistema guarda la película y muestra un mensaje de grabado satisfactoriamente.
Escenario alternativo	<ol style="list-style-type: none"> 6. Los datos introducidos no son válidos. 7. El sistema devuelve un error de validación. 8. El sistema vuelve a la pantalla de Alta-Película

CU_8 ELIMINAR UNA PELÍCULA	
Actor Principal	Administrador
Objetivos	Eliminar una película en la Base de Datos
Pre-condiciones	Estar logado como Administrador La película existe en el sistema
Post-condiciones	La película se elimina de la Base de Datos.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con la relación de películas y con el botón Eliminar-Película. 2. El usuario pulsa el botón Eliminar-Película. 3. El sistema muestra una página con los datos de la película que queremos eliminar. 4. El usuario pulsa el botón Eliminar-Película. 5. El sistema elimina la película 6. El sistema muestra la lista de películas.
Escenario alternativo	<ol style="list-style-type: none"> 5. El sistema da un error de acceso a base de datos. 6. El sistema vuelve a la pantalla con los datos de la película a eliminar.

CU_9 MODIFICAR UNA PELÍCULA	
Actor Principal	Administrador
Objetivos	Modificar una película guardada en Base de Datos
Pre-condiciones	Estar logado como Administrador La película existe en el sistema
Post-condiciones	La película se modifica en Base de Datos.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con la relación de películas y el botón Modificar-Película. 2. El usuario pulsa el botón Modificar-Película. 3. El sistema muestra una página con los datos de la película que queremos Modificar 4. El usuario modifica los datos. 5. El usuario pulsa el botón Guardar. 6. El sistema guarda la película en BBDD 7. El sistema muestra la lista de cines.
Escenario alternativo	<ol style="list-style-type: none"> 6. El sistema da un error con los datos introducidos no son correctos. 7. El sistema vuelve a la pantalla con los datos del cine a modificar. 8. El sistema marca en rojo los datos erróneos.

CU_10		EVALUAR UNA PELÍCULA
Actor Principal	Administrador, Usuario	
Objetivos	Dar una valoración a una película	
Pre-condiciones	Estar logado en el sistema La película existe en el sistema	
Post-condiciones	Se muestra la valoración media en la lista de películas.	
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con la lista de las películas y con el botón Evaluar-Película. 2. El usuario pulsa el botón Evaluar-Película. 3. El sistema muestra una página con los atributos vacíos para evaluar una película. 4. El usuario introduce los valores solicitados. 5. El usuario pulsa el botón Evaluar. 6. El Sistema guarda en BBDD los datos. 7. El Sistema calcula la valoración media de la película. 8. El Sistema muestra la lista de películas y muestra la valoración media. 	
Escenario alternativo	<ol style="list-style-type: none"> 3. El sistema muestra los atributos rellenos con la valoración realizada anteriormente. 4. El usuario cambia la valoración. 5. Volver al paso 5 del escenario principal. 	
Escenario alternativo	<ol style="list-style-type: none"> 6. El sistema da un error por introducir una valoración superior al valor admitido. 7. El Sistema vuelve a la pantalla de datos de la valoración y marca en rojo el campo erróneo. 8. El usuario introduce un valor correcto. 9. Volver al paso 5 el escenario principal. 	

CU_11		VER DETALLE DE UNA PELÍCULA
Actor Principal	Administrador, Usuario	
Objetivos	Consultar el detalle de una película guardada en Base de Datos	
Pre-condiciones	Estar logado en el sistema La película existe en el sistema	
Post-condiciones	Se muestra una página con el detalle de la película.	
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una página con la lista de películas y el botón Ver-Detalle-Película. 2. El usuario pulsa el botón Ver-Detalle-Película. 3. El sistema muestra una página con los datos de la película que queremos Ver. 	
Escenario alternativo	<ol style="list-style-type: none"> 3. El sistema da un error de acceso a la base de datos. 4. El sistema vuelve a la pantalla con la lista de películas para poder pulsar el botón Ver-Detalla-Película. 	

CU_12 BUSCAR UNA PELÍCULA APLICANDO FILTROS	
Actor Principal	Administrador, Usuario
Objetivos	Ver la lista con las películas filtradas
Pre-condiciones	Estar logado en el sistema
Post-condiciones	Se muestra una página con la lista de películas filtradas.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una pestaña para realizar la búsqueda de película aplicando filtros. 2. El usuario pulsa en la pestaña Buscar-Película. 3. El sistema muestra una página con los atributos que puede aplicar un filtro. 4. El usuario introduce los datos a buscar. 5. El usuario pulsa el botón Buscar. 6. El sistema muestra la lista de películas filtradas.
Escenario alternativo	<ol style="list-style-type: none"> 6. El sistema no encuentra datos. 7. El sistema muestra una lista vacía.

CU_13 VER EVALUACION DE UNA PELICULA	
Actor Principal	Administrador, Usuario
Objetivos	Ver lista de las películas con la evaluación media.
Pre-condiciones	Estar logado en el sistema. Se ha añadido al menos una evaluación a la película.
Post-condiciones	Se muestra la lista de películas con su evaluación media.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una pestaña para ver las películas. 2. El usuario pulsa en la pestaña Películas. 3. El Sistema muestra una página con la lista de las películas. 4. El Sistema muestra la media de cada película.
Escenario alternativo	<ol style="list-style-type: none"> 4. El sistema muestra 0,0 en las que no tengan evaluación previa.

CU_14 VER EVALUACIONES Y COMENTARIOS DE UNA PELICULA	
Actor Principal	Administrador
Objetivos	Ver lista de las evaluaciones y comentarios de la película seleccionada.
Pre-condiciones	Estar logado en el sistema como Administrador.
Post-condiciones	Se muestra la lista de evaluaciones y comentarios de la película seleccionada.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una pestaña para ver las evaluaciones y comentarios de las películas. 2. El usuario pulsa en la pestaña. 3. El Sistema muestra una página para seleccionar la película. 4. El usuario selecciona una película. 5. El usuario pulsa el botón Seleccionar 6. El Sistema muestra la lista con los comentarios y evaluaciones de la película seleccionada.
Escenario alternativo	<ol style="list-style-type: none"> 6. El sistema no tiene datos de la película seleccionada 7. El sistema saca una lista vacía.

CU_15 ELIMINAR EVALUACIONES Y COMENTARIOS DE UNA PELICULA	
Actor Principal	Administrador
Objetivos	Eliminar las evaluaciones y sus comentarios de BBDD.
Pre-condiciones	Estar logado en el sistema como Administrador. Al menos hay una evaluación en el sistema
Post-condiciones	Se elimina de BBDD la evaluación y sus comentarios.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una lista con todas las evaluaciones y comentarios. 2. El usuario pulsa en el botón ELIMINAR. 3. El Sistema muestra una página con los datos que se quieren eliminar. 4. El usuario pulsa CONFIRMAR. 5. El Sistema elimina los datos de la BBDD. 6. El Sistema muestra la lista con las evaluaciones y comentarios.
Escenario alternativo	<ol style="list-style-type: none"> 5. El sistema da error de acceso a la BBDD. 6. El sistema muestra la lista con las evaluaciones y los comentarios.

CU_16	VER TODAS LAS EVALUACIONES Y COMENTARIOS
Actor Principal	Administrador
Objetivos	Ver todas las evaluaciones y sus comentarios.
Pre-condiciones	Estar logado en el sistema como Administrador. Al menos hay una evaluación en el sistema
Post-condiciones	Se muestra una lista con todas las evaluaciones y sus comentarios.
Escenario principal	<ol style="list-style-type: none"> 1. El Sistema muestra una pestaña Listar-Evaluaciones. 2. El usuario pulsa en la pestaña Listar-Evaluaciones. 3. El Sistema muestra una página con la lista de las evaluaciones y sus comentarios.

2.5. Diagrama de Casos de Uso

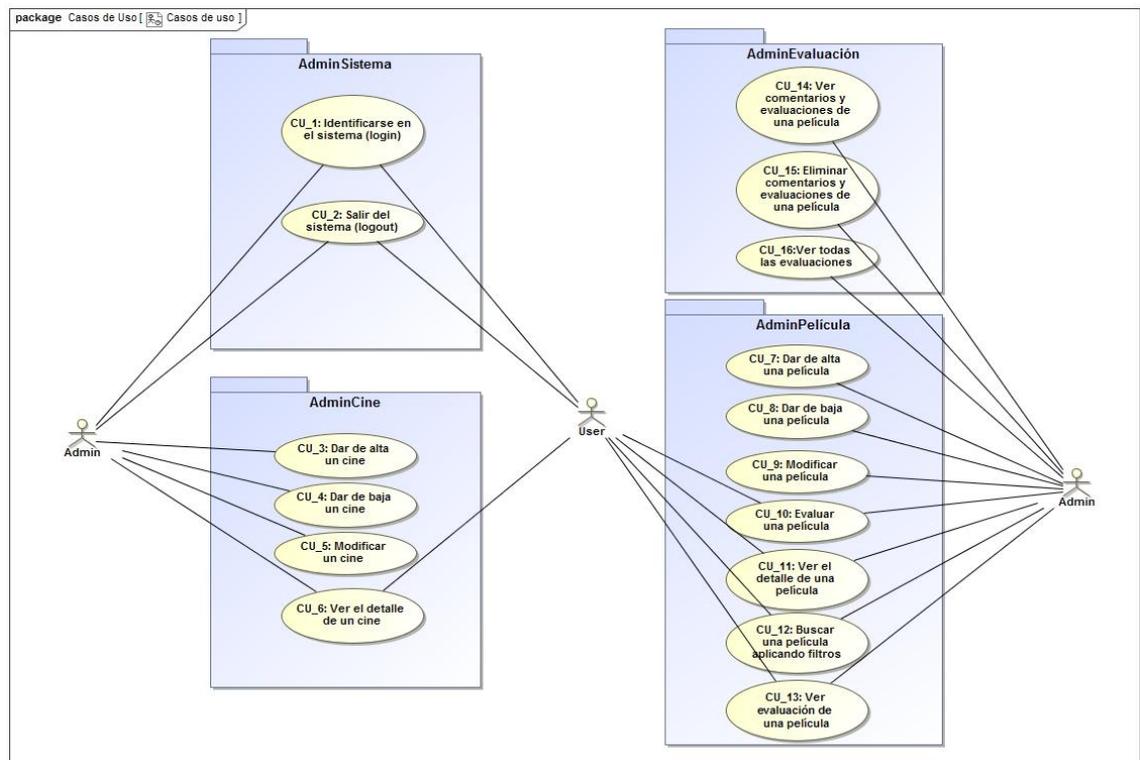


Ilustración 3 - Diagrama de Casos de Uso

Nota: Los roles Admin y User se corresponden con Administrador y Usuario respectivamente. He duplicado el rol Admin para mejorar la visualización

2.6. Diagrama de clases

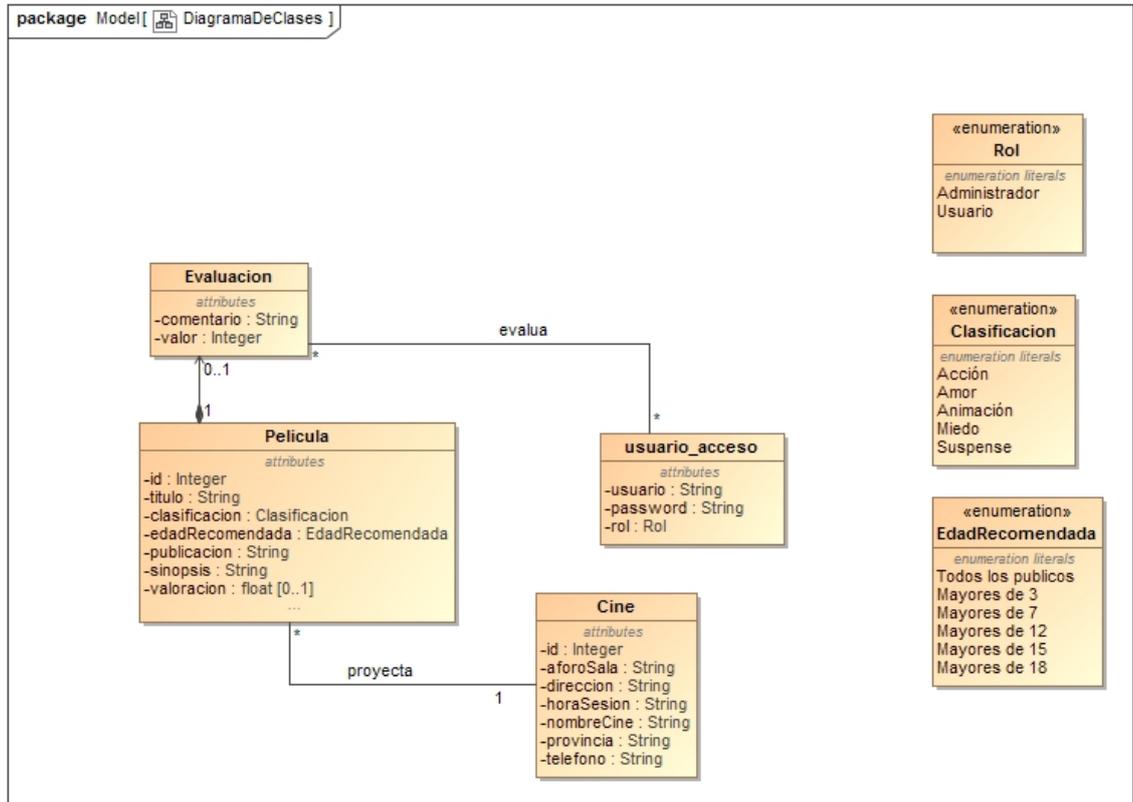


Ilustración 4 - Diagrama de Clases

Nota: Las clases que he incluido en el diagrama anterior del tipo *enumeration* no las voy a considerar a la hora de hacer el modelo de datos. Voy a controlarlos por código como constantes.

2.7. Diseño relacional de la base de datos

El Sistema de Gestión de Base de datos con el que voy a implementar el diseño es PostgreSQL y el diseño relacional de las tablas es el siguiente:

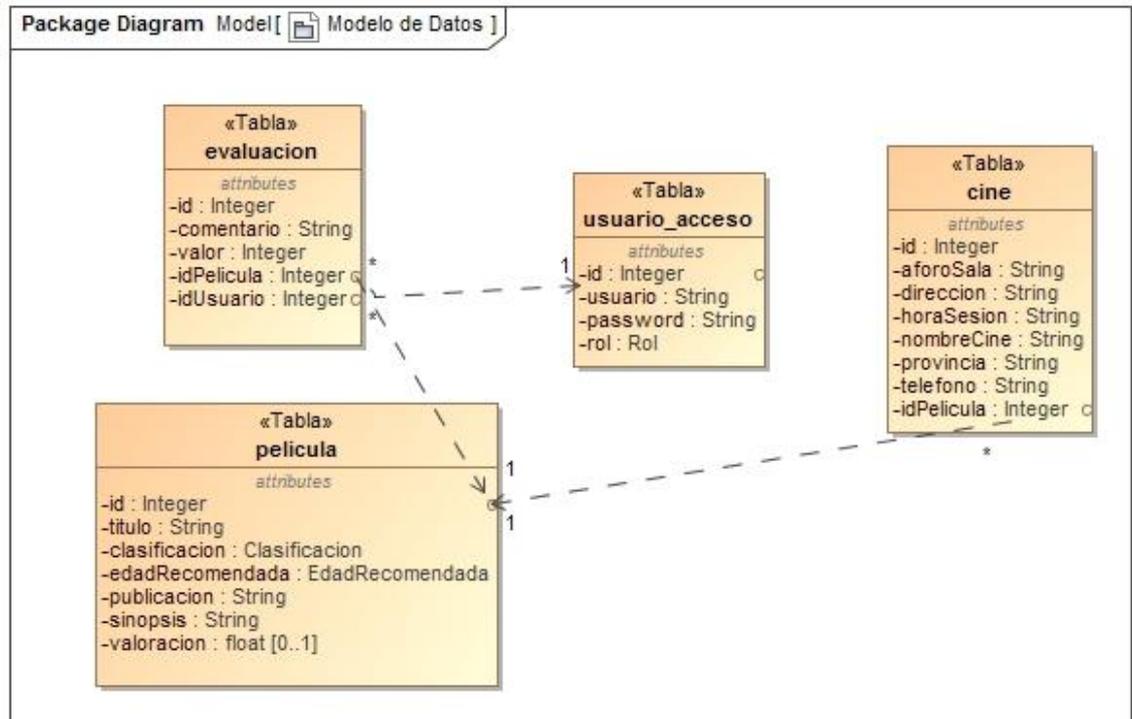


Ilustración 5 – Diseño relacional de la BBDD

Nota: He definido para el atributo password el tipo de datos como “String” y quizá se debería implementar un tipo de datos específico para este tipo de dato. No he querido reflejarlo en el diseño de BBDD por implementarlo en el código. También los atributos Teléfono y dirección de la tabla Cine se podría implementar un tipo específico, pero para este proyecto no lo veo necesario.

Con respecto a las relaciones, un usuario puede realizar varias evaluaciones, una película puede tener varias evaluaciones y una película puede mostrarse en varios cines pero cada cine sólo puede publicar una película en esta sesión.

2.8. Diagrama de componentes

El patrón seleccionado para el diseño de componentes, es el Modelo Vista Controlador (MVC). Teniendo en cuenta este patrón, desde el punto de vista de la computación, este sería el diseño:

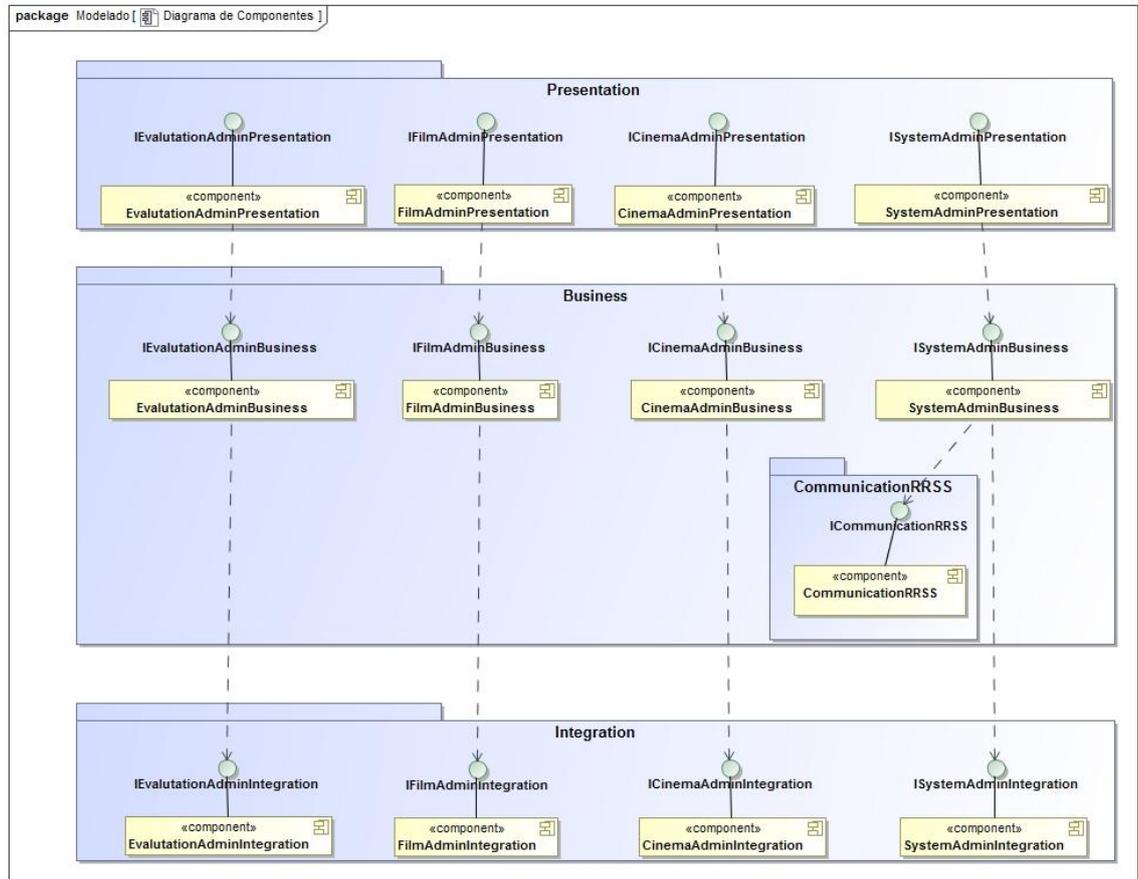


Ilustración 6 – Diagrama de componentes

Notas: He añadido en la capa de Negocio una posible integración con Redes Sociales para permitir en un futuro publicar información automáticamente en las RRSS (Facebook, Twitter,...).

Para ampliar el diagrama anterior, voy a añadir un diagrama por cada capa indicando la interfaz de los componentes y sus operaciones.

Capa Presentación:

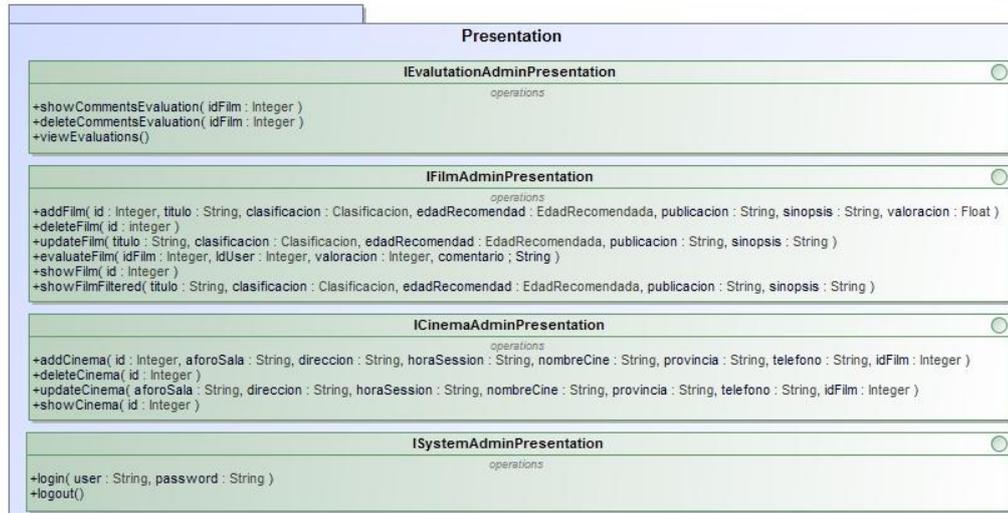


Ilustración 7 -Interfaz Componentes Capa Presentación

Capa Negocio:

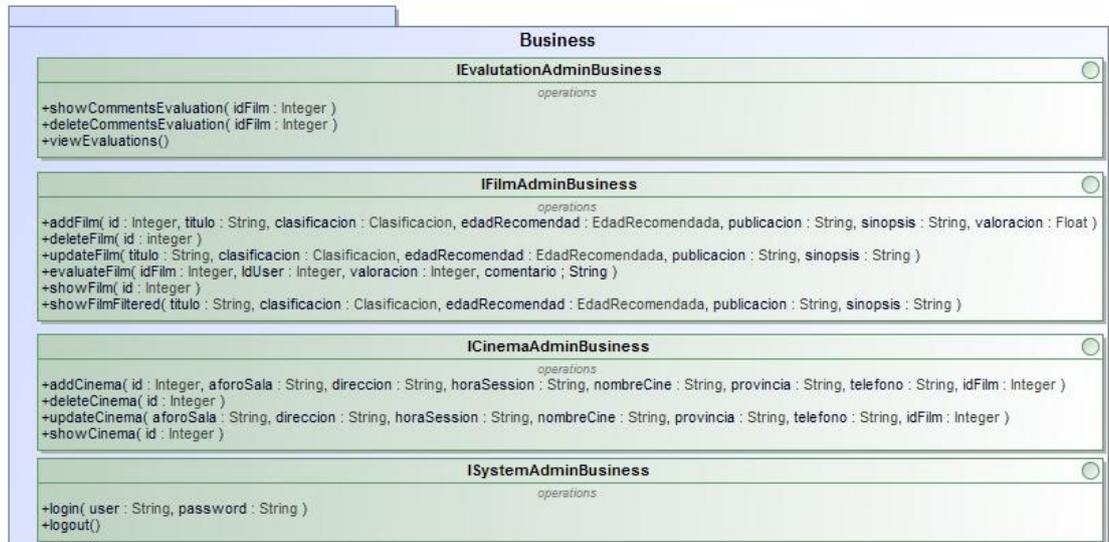


Ilustración 8 – Interfaz Componentes Capa Negocio

Capa Integración o Datos:

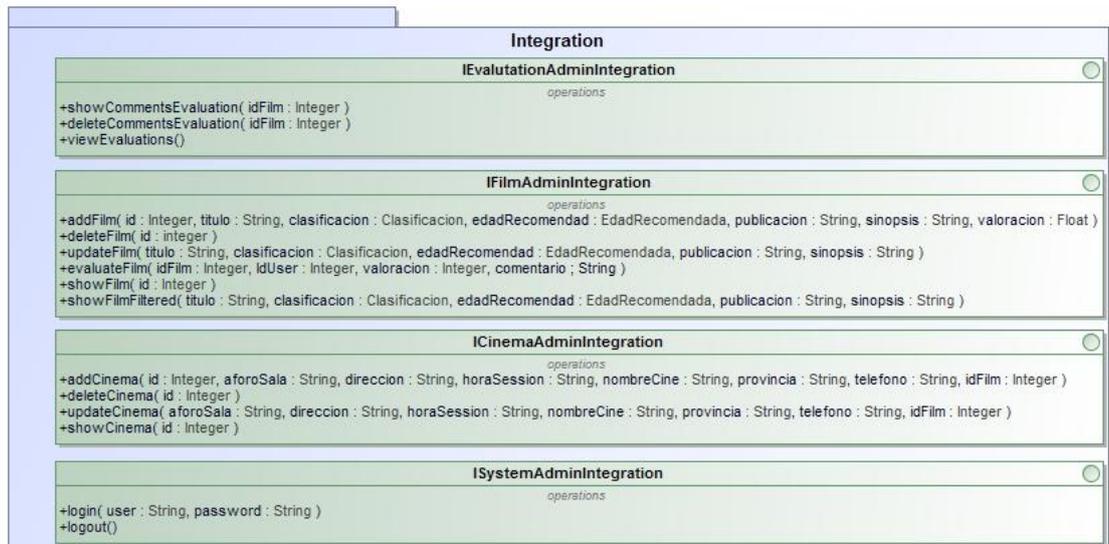


Ilustración 9 - Interfaz Componentes Capa Integración

2.9. Diagrama de arquitectura

Tras el análisis realizado, uno de los requisitos no funcionales del sistema es el uso de la tecnología JavaEE. Debido a este requisito, veo preciso el uso de Spring como Framework del paradigma JavaEE. En la parte del cliente será utilizado un navegador Web como visor de la aplicación.

Para el diseño arquitectónico, partimos del diseño de componentes visto en el punto anterior.

Realizaremos los siguientes pasos:

- 1) Refinamiento de los componentes
- 2) Diseño del componente y aplicación del perfil JavaEE.

Cómo el patrón que he seleccionado de arquitectura de software es el Modelo Vista Controlador, voy a realizar los pasos anteriores diferenciados por cada una de las tres capas (Presentación, Negocio e Integración)

2.9.1. Capa de Presentación

Refinamiento de los componentes

Según hemos definido, las operaciones que tenemos que desarrollar por componente son las siguientes:

Componente *EvaluationAdmin*

- ShowCommentsEvaluation → ver los comentarios y las evaluaciones de una película
- DeleteCommentsEvaluation → Borrar los comentarios y evaluaciones seleccionados

- ViewEvaluations → Ver evaluaciones

Componente FilmAdmin

- AddFilm → Añadir una película
- DeleteFilm → Borrar una película
- UpdateFilm → Actualizar una película
- EvaluateFilm → Evaluar una película
- ShowFilm → Mostrar los detalles de una película
- ShowFilmFiltered → Mostrar la lista de películas aplicando un filtro

Componente CinemaAdmin

- AddCinema → Añadir Cine
- DeleteCinema → Borrar Cine
- UpdateCinema → Actualizar Cine
- ShowCinema → Ver detalle del cine

Componente SystemAdmin

- Login → Logarse en el Sistema
- Logout → Desconectarse del Sistema

El patrón que he decidido usar para la arquitectura de esta aplicación es el Modelo Vista Controlador (MVC). El primer refinamiento que podemos hacer es separar el componente que hará funciones de controlador del que generará las vistas. Para la capa de presentación, este será el refinamiento:

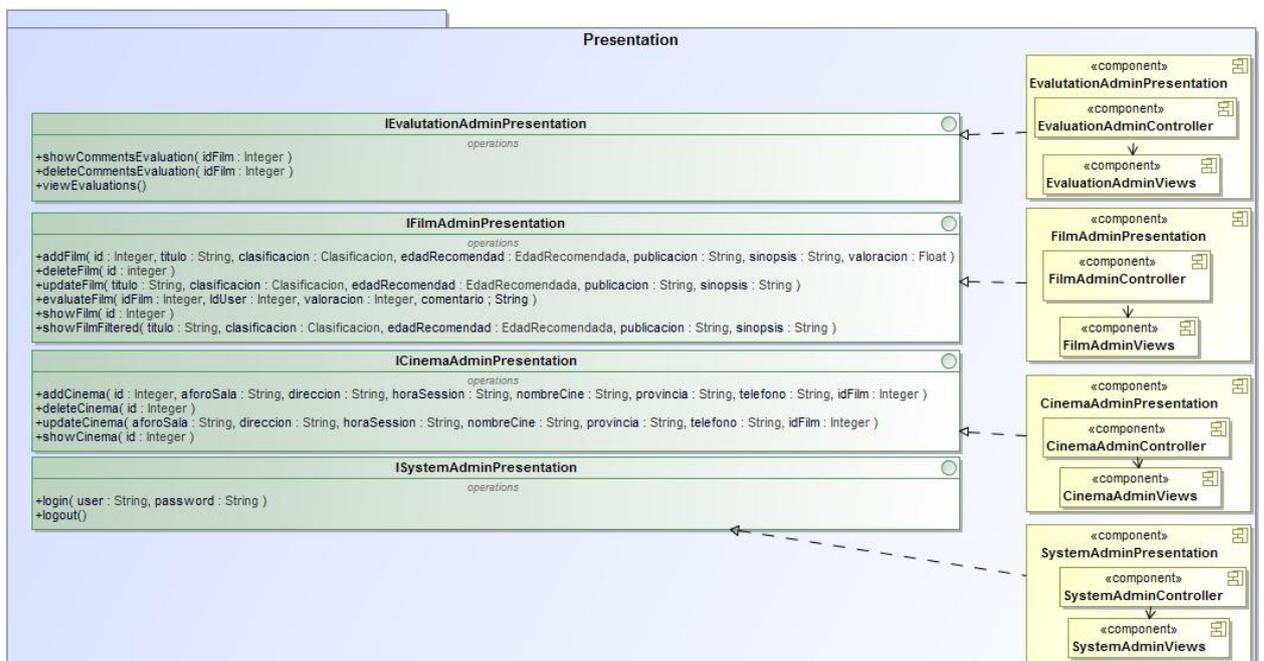


Ilustración 10 - Refinamiento - Presentación

Diseño del componente

Una vez refinados los componentes de la capa de presentación, voy a proponer las siguientes decisiones de diseño:

1. Habrá un Controller por componente, agrupando cada una de las operaciones que tiene que realizar. Para ello, voy a implementar el patrón FrontController. Las operaciones no se implementarán dentro del controlador, voy a utilizar el patrón Command para poder hacerlo. Con este patrón hacemos responsable de coordinar las operaciones al Controller, pero desacoplamos la implementación. Cada acción que puede realizar un usuario, estará mapeada con una vista, y será el controlador el que ejecuta estas acciones.
2. Vistas a implementar:

Componente EvaluationAdmin

- ShowCommentsEvaluationView → Muestra las evaluaciones y los comentarios de una película seleccionada.
- DeleteCommentsEvaluationView → Muestra el comentarios y la evaluación que se va a eliminar.
- ShowEvaluationsView → Muestra una lista con todas las evaluaciones.

Componente FilmAdmin

- AddFilmView → Muestra los campos de alta de una película.
- DeleteFilmView → Muestra los datos de la película que se va a eliminar.
- UpdateFilmView → Muestra los datos de la película para modificar.
- EvaluateFilmView → Muestra la vista para evaluar una película y poder poner comentarios.
- ShowFilmView → Muestra todos los datos de la película.
- ShowFilmFilteredView → Muestra una vista con los datos que podemos filtrar.
- ShowFilmsListView → Muestra una lista de películas.

Componente CinemaAdmin

- AddCinemaView → Muestra una vista para añadir un cine.
- DeleteCinemaView → Muestra los datos del cine que vamos a eliminar, para confirmar su eliminación.
- UpdateCinemaView → Muestra los datos del cine que podemos modificar.
- ShowCinemaView → Muestra todos los datos del cine.
- ShowCinemaListView → Muestra una lista de cines.

Componente SystemAdmin

- LoginView → Muestra la vista para logarse en el Sistema.
- ErrView → Vista general de cualquier error.

Notas:

- He quitado la vista que muestra la evaluación por considerar que la evaluación será mostrada junto con la lista de películas.
- He añadido una lista tanto de películas como en cines, que no había contemplado pero que creo necesario.
- También he creado una vista que contemplará los errores de la aplicación.
- La pantalla de Logout no muestra ninguna vista nueva. Redirigirá a la de Login
- Una vez hecho login, se verá una vista con la lista de los cines.

3. Acciones :

Componente EvaluationAdmin

- ShowCommentsEvaluation → ver los comentarios y las evaluaciones de una película
- DeleteCommentsEvaluation → Borrar los comentarios y evaluaciones seleccionados
- ViewEvaluations → Ver evaluaciones

Componente FilmAdmin

- AddFilm → Añadir una película
- DeleteFilm → Borrar una película
- UpdateFilm → Actualizar una película
- EvaluateFilm → Evaluar una película
- ShowFilm → Mostrar los detalles de una película
- ShowFilmFiltered → Mostrar la lista de películas aplicando un filtro

Componente CinemaAdmin

- AddCinema → Añadir Cine
- DeleteCinema → Borrar Cine
- UpdateCinema → Actualizar Cine
- ShowCinema → Ver detalle del cine

Componente SystemAdmin

- Login → Logar un usuario en el Sistema
- Logout → Desconecta un usuario del Sistema

Si aplicamos los criterios mencionados, obtenemos el siguiente diseño:

EvaluationAdminPresentation

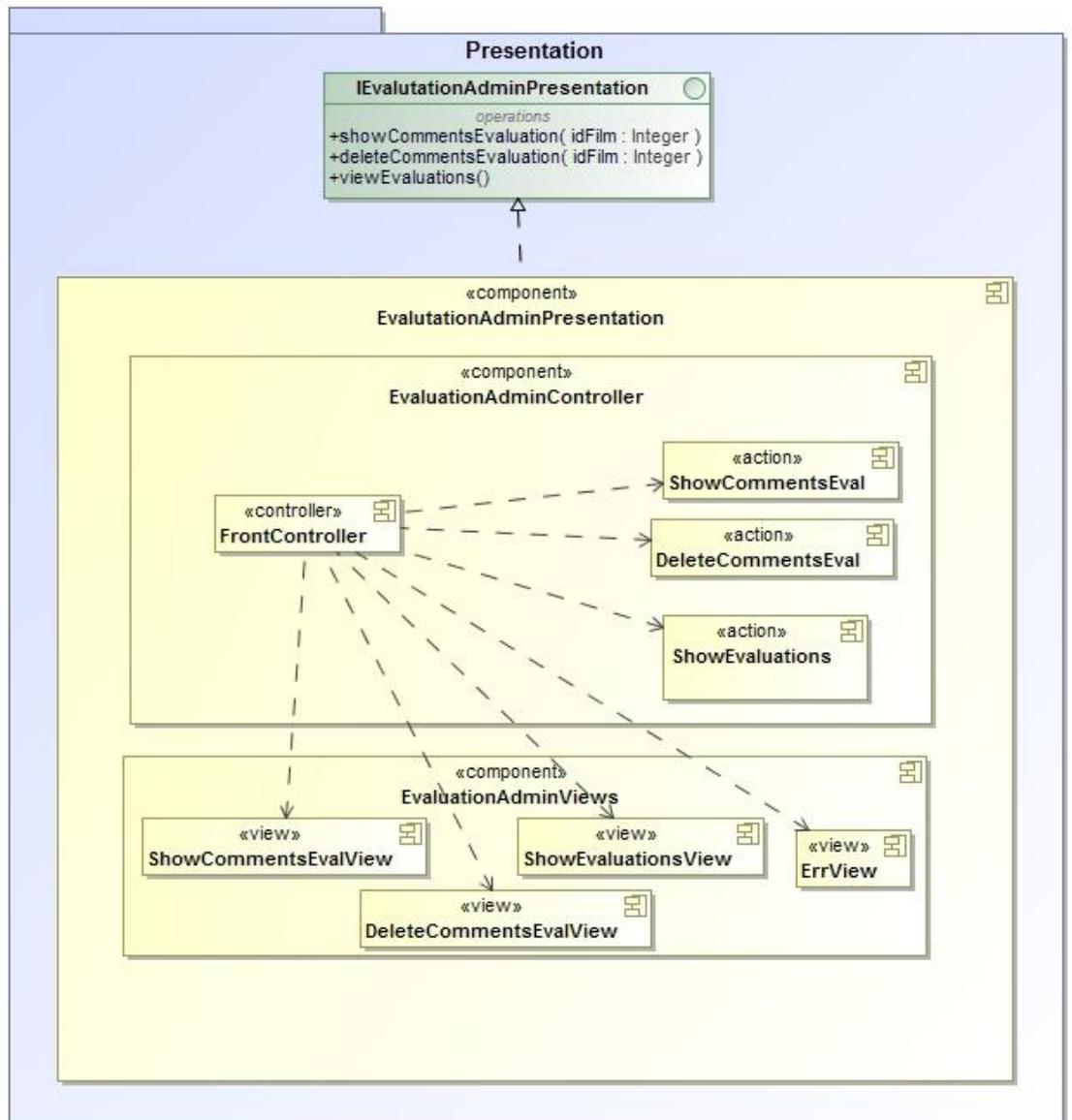


Ilustración 11 – Diseño Componente Evaluacion Refinado (Presentacion)

FilmAdminPresentation

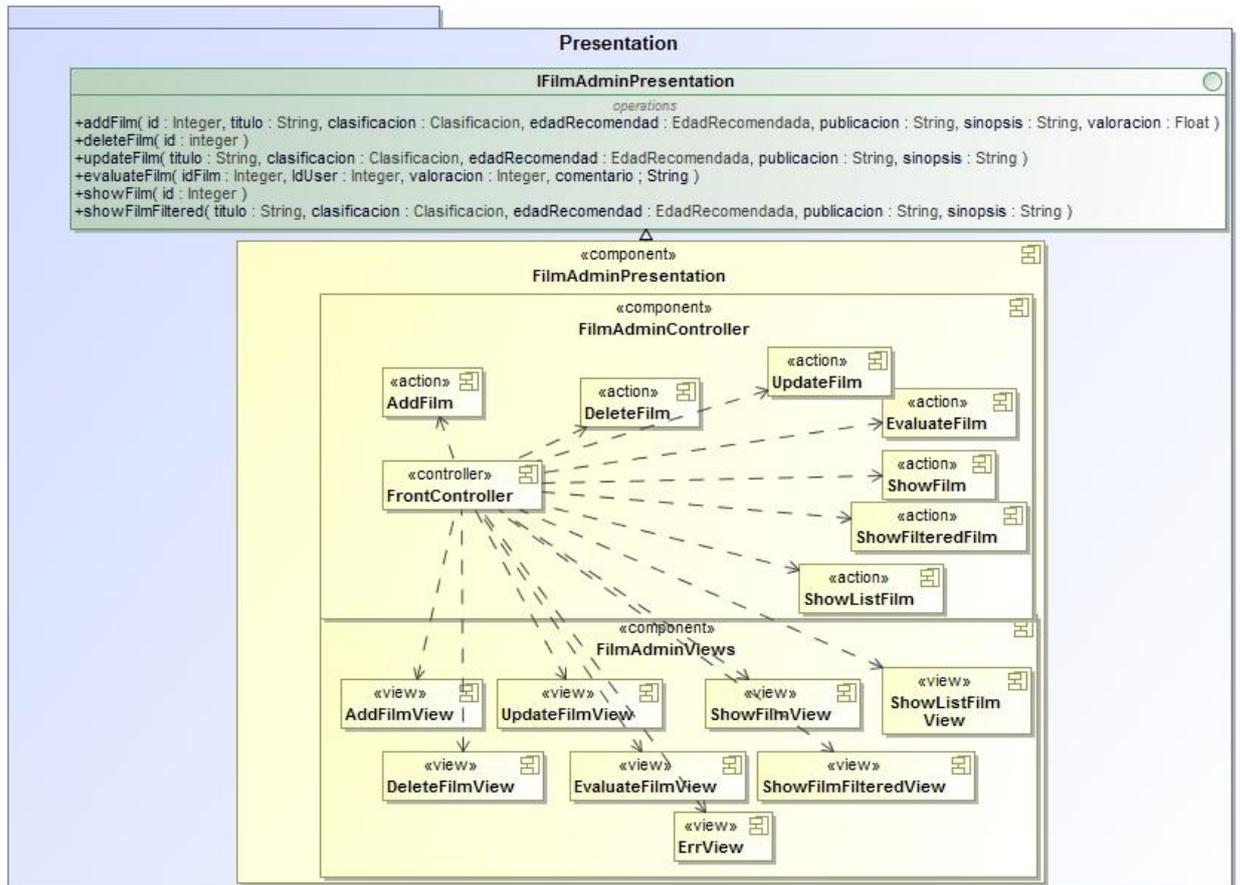


Ilustración 12 - Diseño Componente FilmAdmin Refinado (Presentacion)

CinemaAdminPresentation

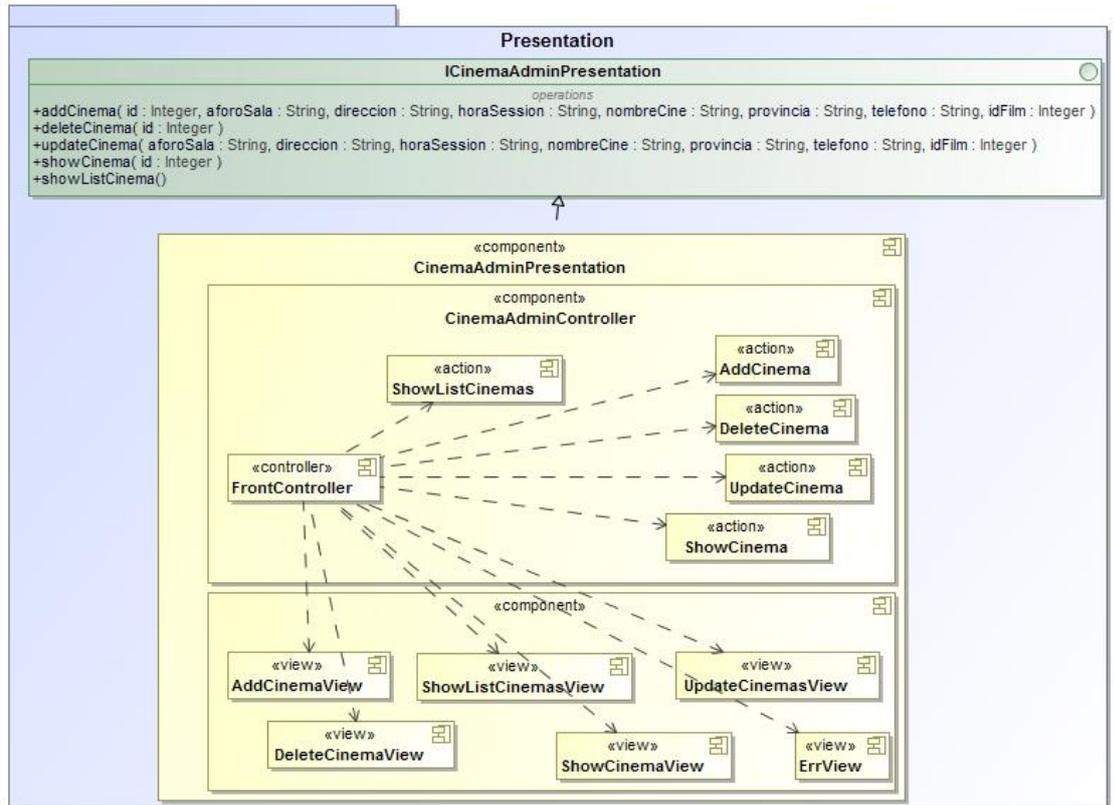


Ilustración 13 - Diseño Componente CinemaAdmin Refinado (Presentacion)

SystemAdminPresentation

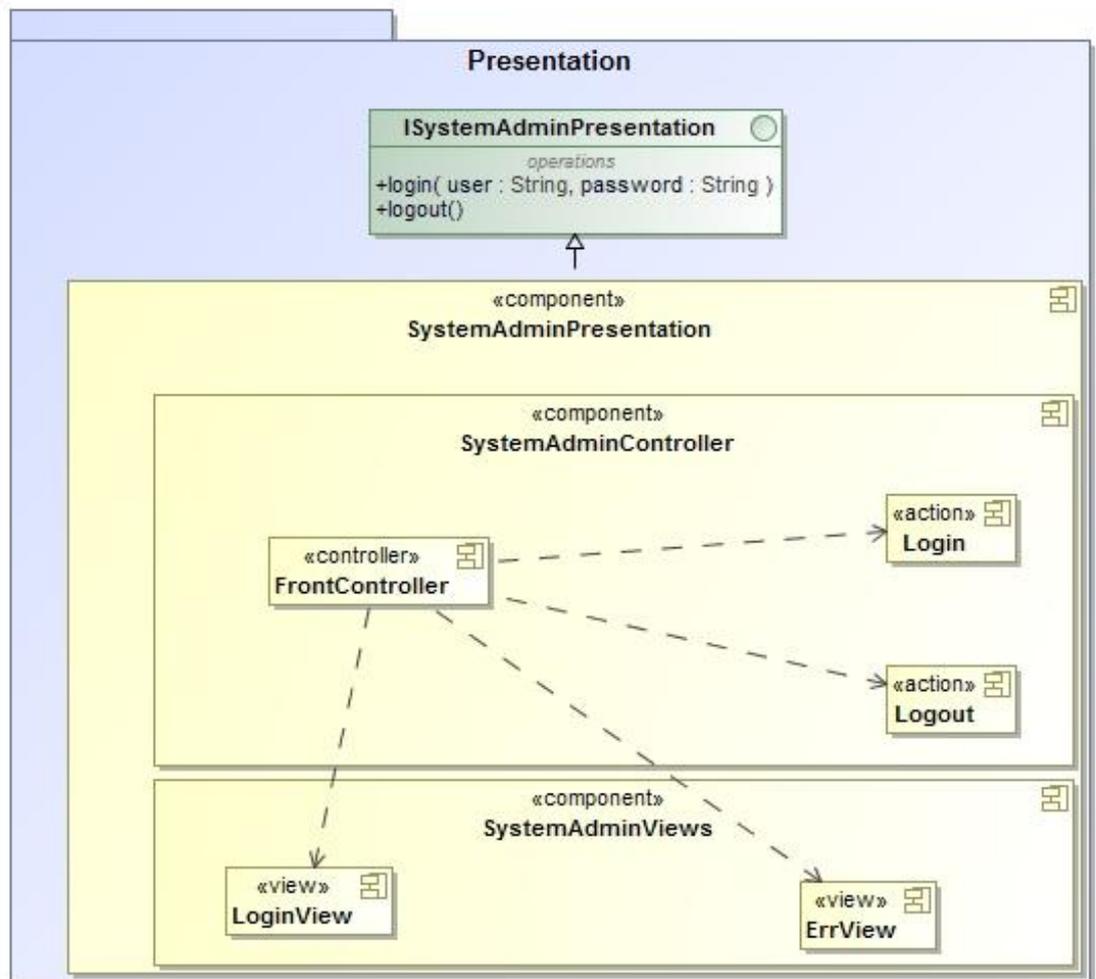


Ilustración 14 - Diseño Componente SystemAdmin Refinado (Presentacion)

Aplicación del perfil JavaEE

Como he comentado, y tras el estudio de los requisitos no funcionales, voy a utilizar tecnología JavaEE. Dentro del paradigma de JavaEE, utilizaré Spring MVC. Para la capa de presentación, Utilizaremos un Controller que es el encargado de llamar a los Services de la capa de negocio y a las vistas. Para la realización de las vistas utilizaré JSP Java Servlet Pages.

EvaluationAdminPresentation

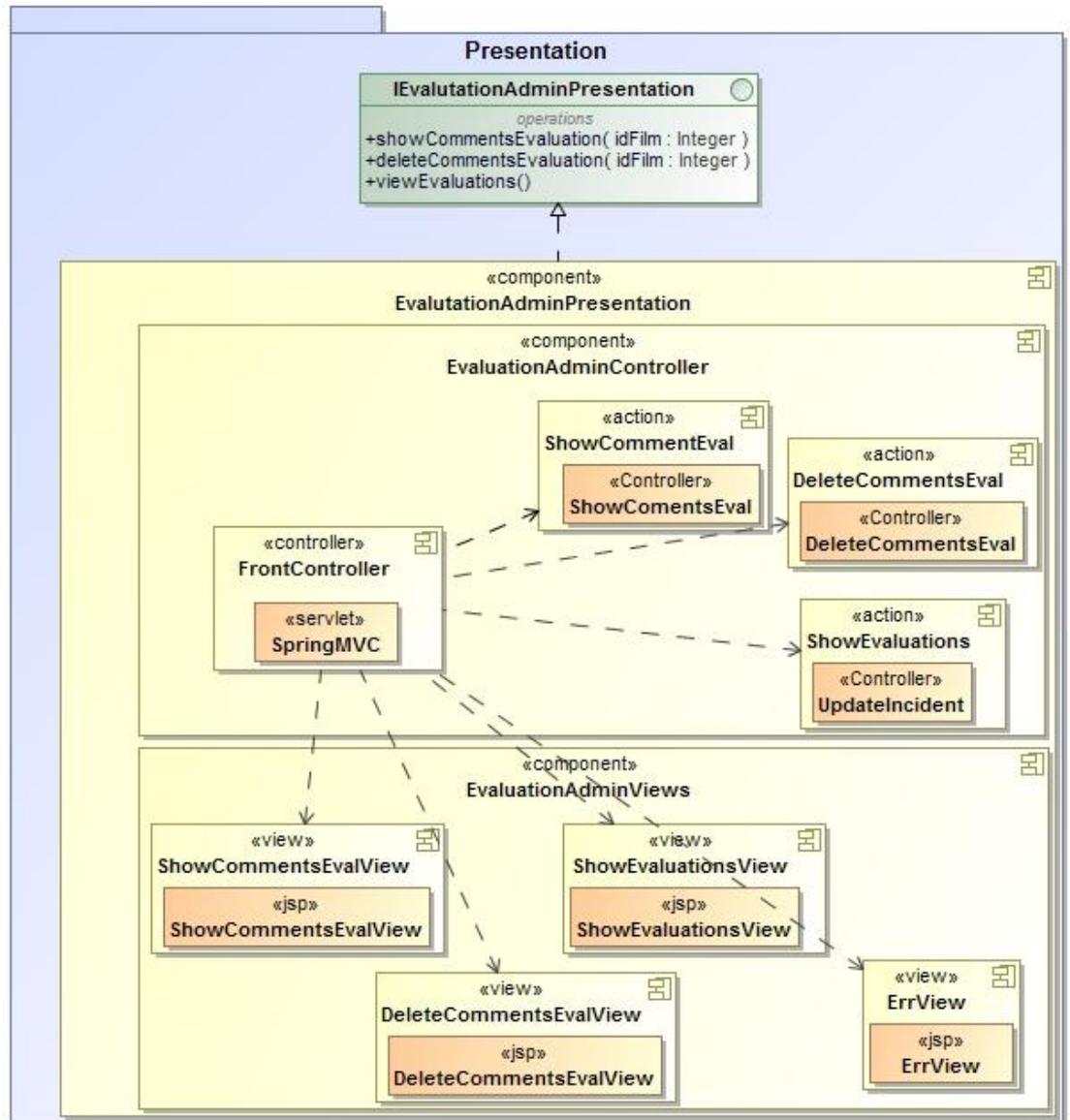


Ilustración 15 - Diseño Componente `EvaluationAdminPresentation` JavaEE

FilmAdminPresentation

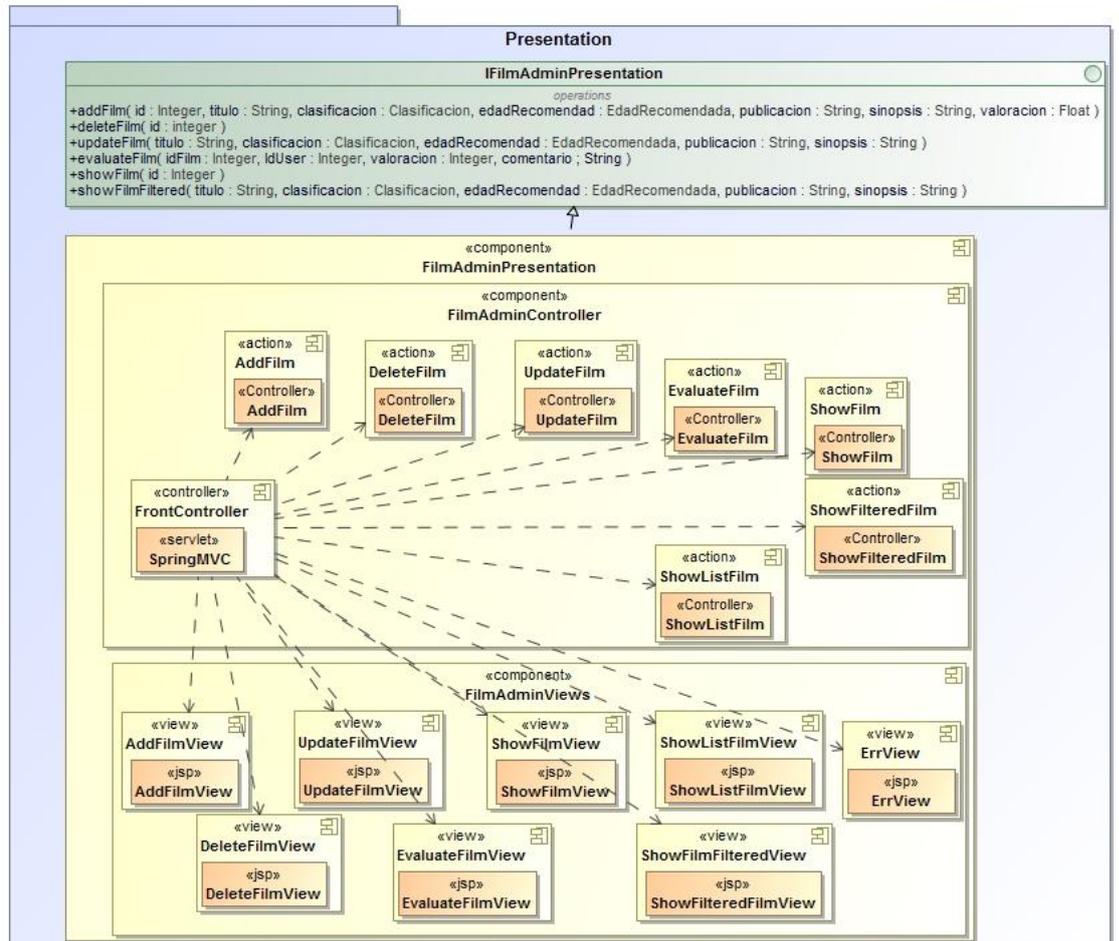


Ilustración 16 - Diseño Componente FilmAdminPresentation JavaEE

CinemaAdminPresentation

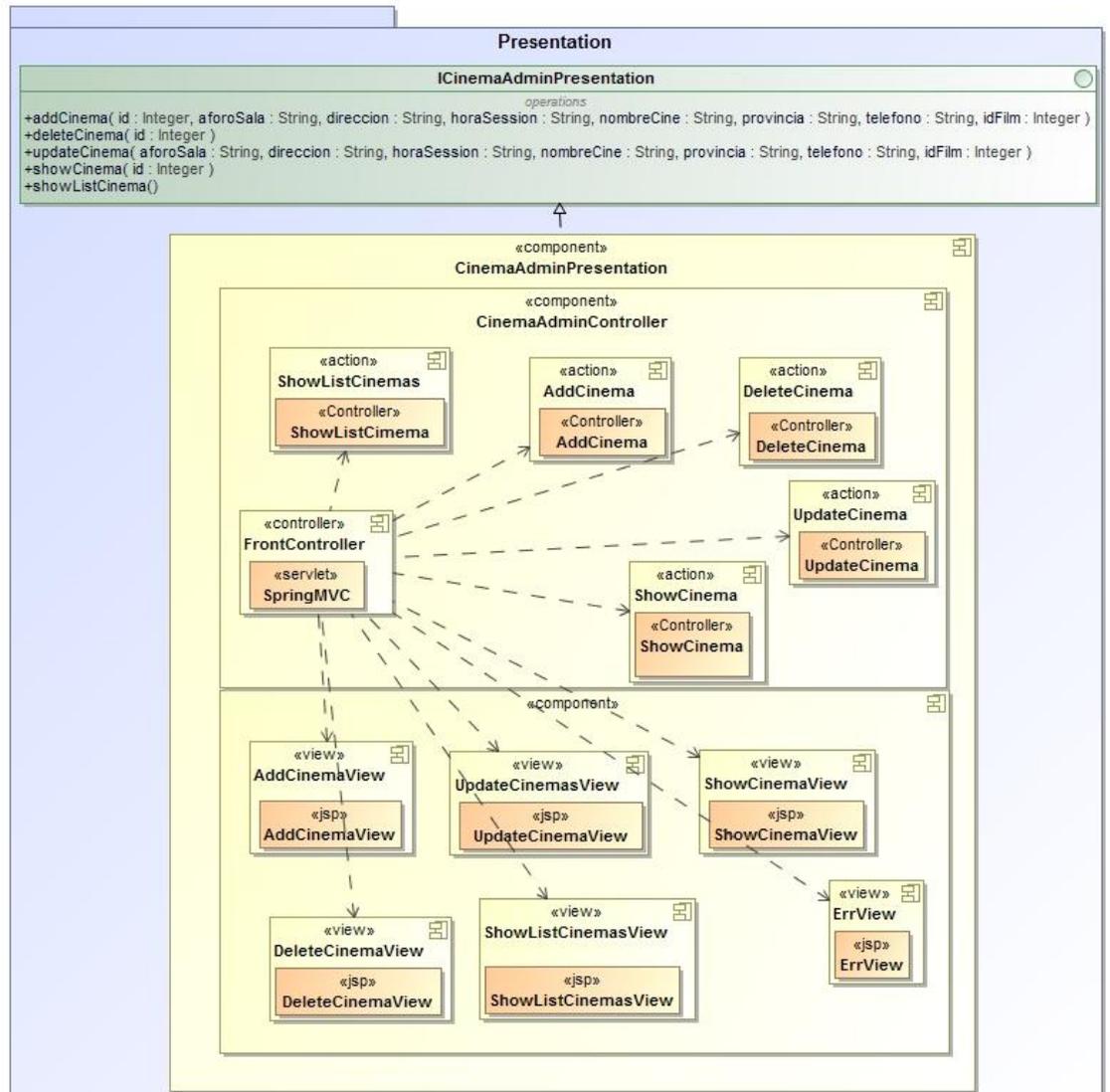


Ilustración 17 - Diseño Componente CinemaAdminPresentation JavaEE

SystemAdminPresentation

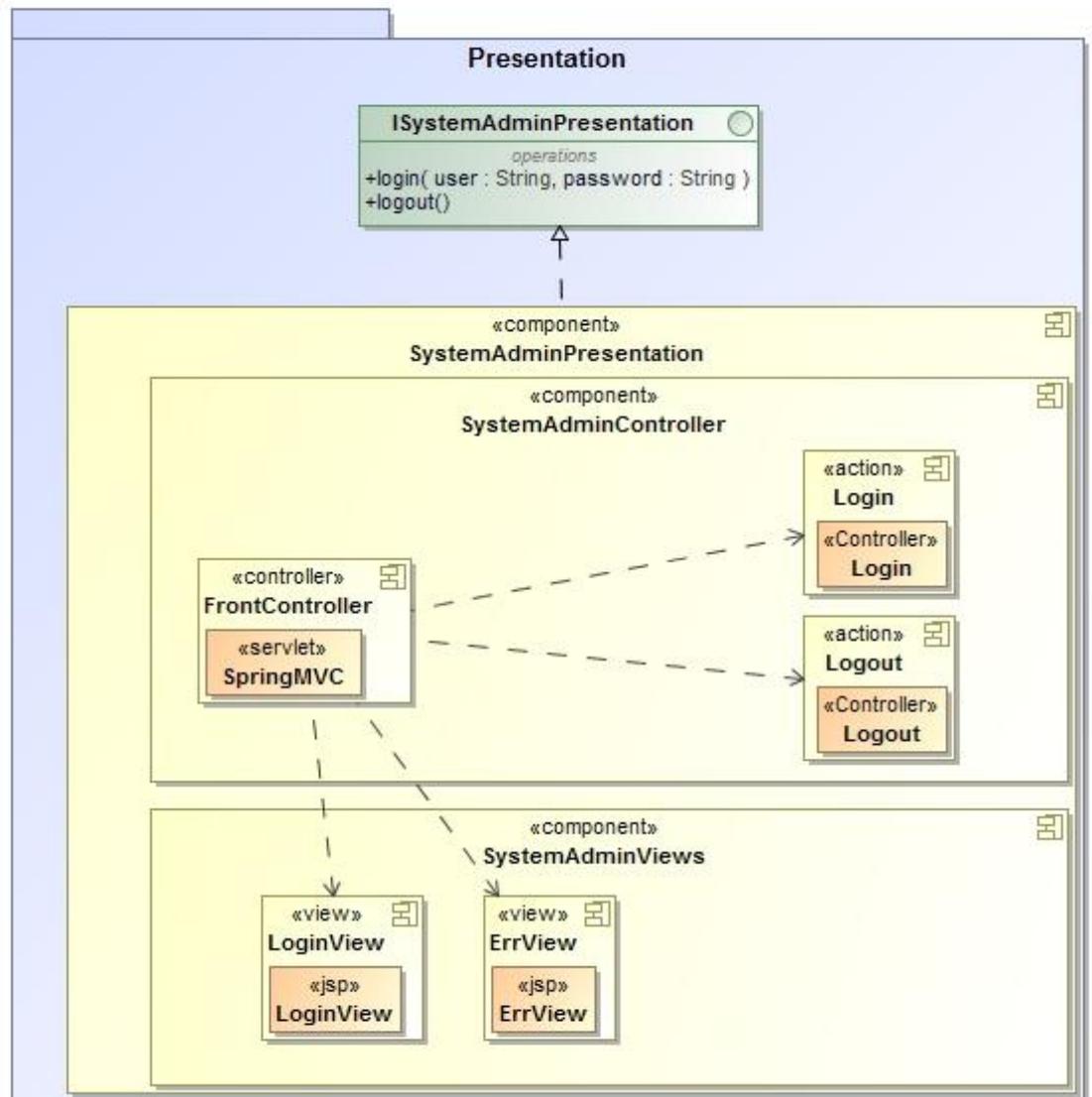


Ilustración 18 - Diseño Componente SystemAdminPresentation JavaEE

2.9.2. Capa de Negocio

Refinamiento de los componentes

La capa de negocio ya se encuentra refinada, por lo que estaría preparada para hacerse tanto el diseño como para aplicar el perfil JavaEE. Vamos a implementar la capa de Negocio usando el patrón Fachada, que va a permitir reducir el acoplamiento entre las capas presentación y negocio, facilitando el acceso a los diferentes subsistemas, ofreciendo un punto de acceso único. Para implementarlo nos remitimos otra vez al Framework Spring para

utilizar Services. Se crearán por un lado las interfaces para el acceso único y por otro la implementación.

Diseño del componente



Ilustración 19 - Diseño de los Componentes para la capa de Negocio.

Aplicación del perfil JavaEE

Como en el diseño hemos contemplado la posibilidad de conectarse a sistemas externos, necesitaríamos dar un acceso local y otro remoto. Para aplicar el perfil JavaEE, vamos a implementar unos Servicios junto con sus Interfaces, utilizando el Framework que nos facilita Spring.

EvaluationAdminBusiness

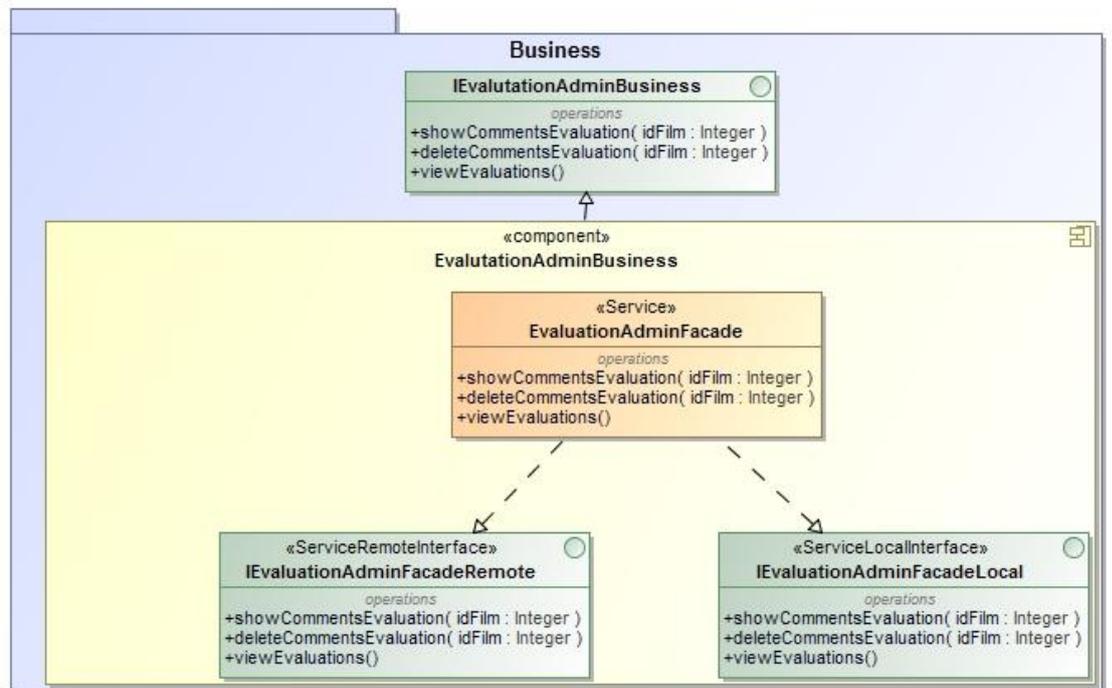


Ilustración 20 - Diagrama Evaluacion capa Negocio JavaEE

FilmAdminBusiness

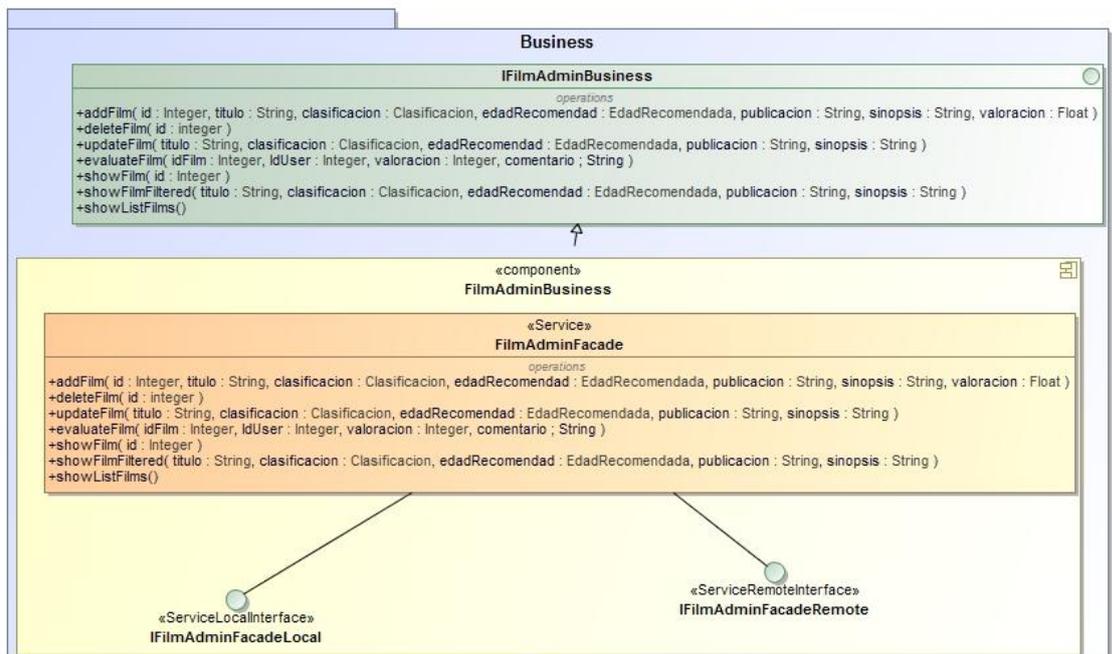


Ilustración 21 – Diagrama película capa Negocio JavaEE

Nota : He quitado las operaciones de las interfaces para los Services Local y Remote para que se visualice mejor en el diagrama.

CinemaAdminBusiness

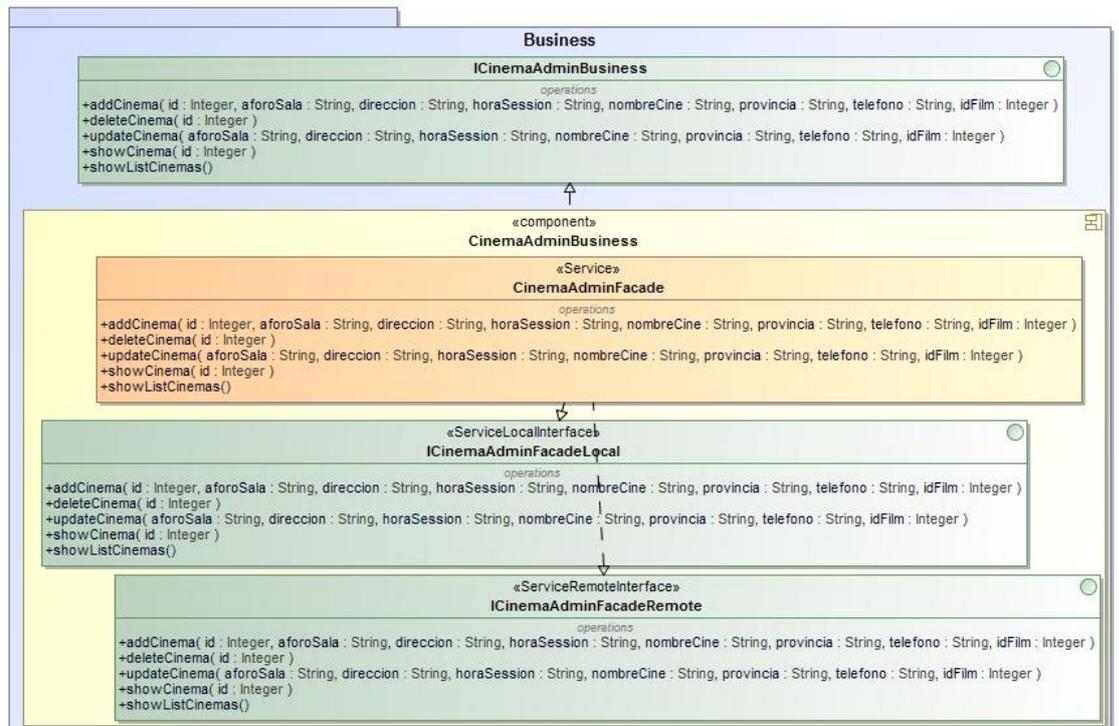


Ilustración 22 - Diagrama cine capa Negocio JavaEE

SystemAdminBusiness

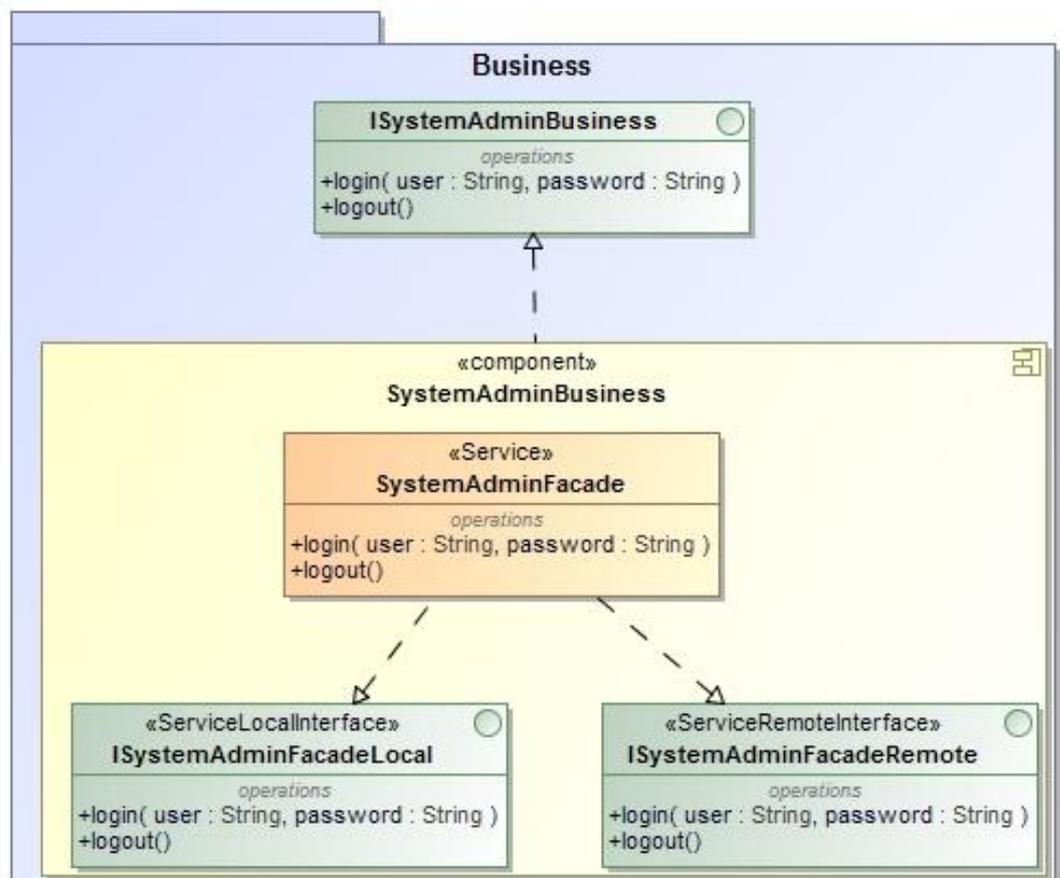


Ilustración 23 - Diagrama SystemAdmin capa de Negocio JavaEE

2.9.3. Capa de Integración

Refinamiento de los componentes

Los datos serán almacenados en una base de datos relacional. El acceso siempre será local desde la capa de negocio.

Diseño del componente

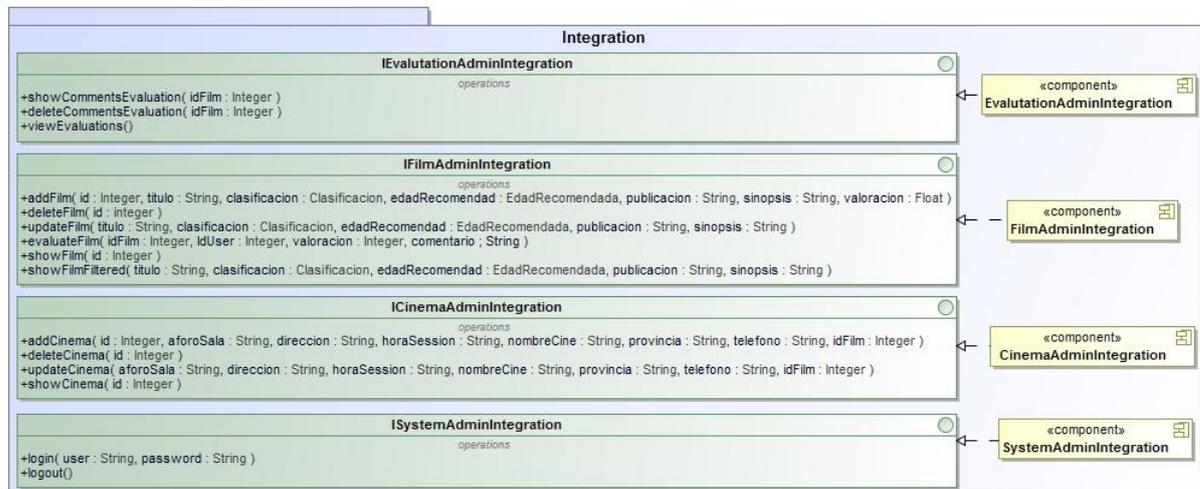


Ilustración 24 - Diagrama de componentes capa Integración

Aplicación del perfil JavaEE

Para la capa de integración, vamos a implementar los componentes usando Spring Data y JPA. Usaremos el Repository para el acceso a datos. Teniendo en cuenta las decisiones de diseño comentadas anteriormente obtenemos el siguiente diagrama para la capa de integración:

EvaluatonAdminIntegration

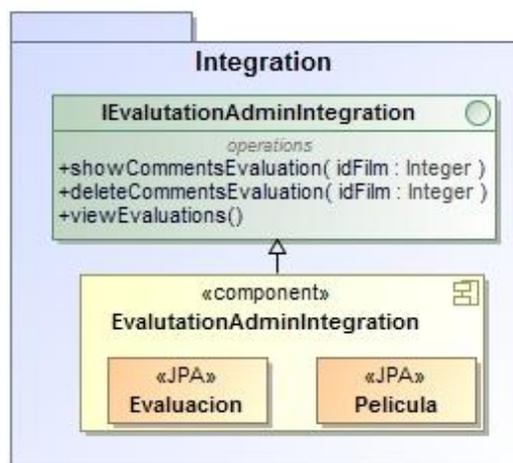


Ilustración 25 - Diagrama EvaluatonAdmin capa Integration JavaEE

FilmAdminIntegration

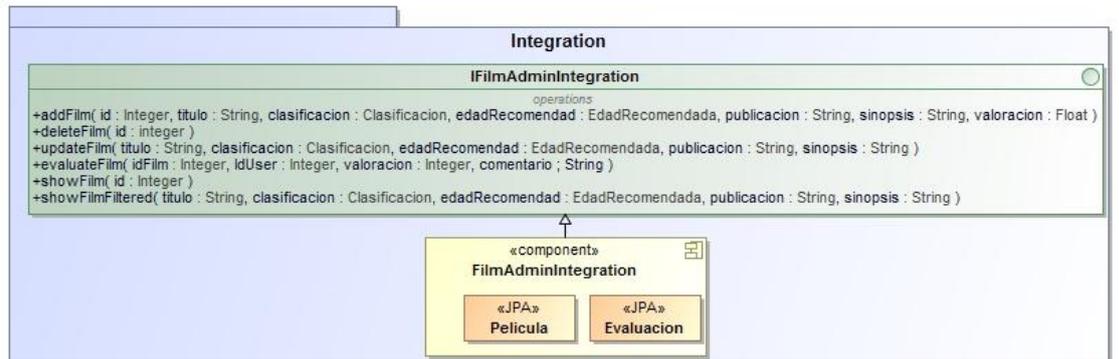


Ilustración 26 - Diagrama FilmAdmin capa Integración JavaEE

CinemaAdminIntegration

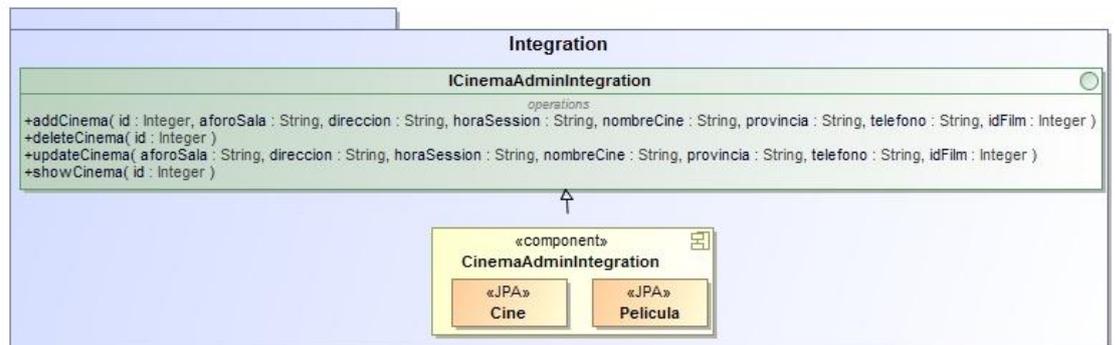


Ilustración 27 - Diagrama CinemaAdmin capa Integración JavaEE

SystemAdminIntegration

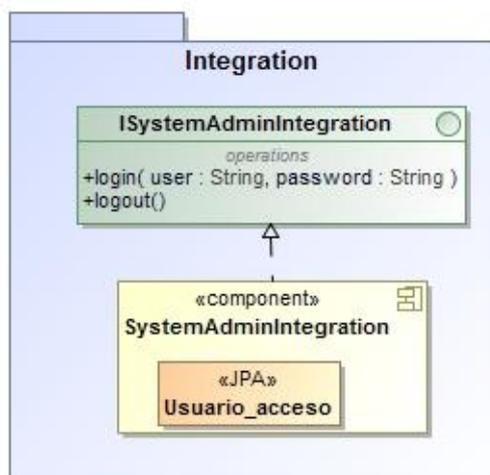


Ilustración 28 - Diagrama SystemAdmin capa Integración JavaEE

Dependencias entre integración y el punto de vista de la información.

Anteriormente hemos podido ver las clases desde el punto de vista de la información. Voy a mostrar el diagrama que relaciona las clases comentadas con los componentes que he desarrollado en el punto anterior:

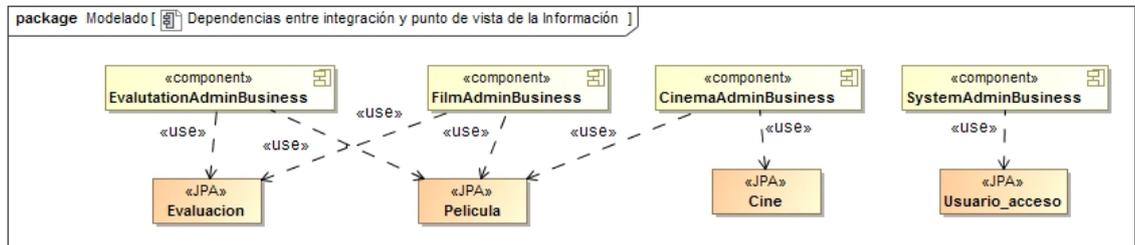


Ilustración 29 - Diagrama de dependencias Integración y el punto de vista de la información

2.10. Arquitectura

Uno de los requisitos de esta aplicación es el desarrollo de una aplicación web en JavaEE. Dado que el paradigma JavaEE abarca muchos componentes, servicios y frameworks, voy a describir las que usaré para el desarrollo de mi TFG.

El cliente será un navegador web. No necesitará instalaciones de ningún tipo, más que el propio navegador que se quiere usar. En mi caso voy a usar el cliente Firefox Quantum 62.0.3.

Con respecto a la capa de presentación o capa web, se realizará mediante el uso de JSP Java Servlet Pages junto con Spring MVC. Como voy a utilizar Spring MVC, usaré los Controller para que hagan las llamadas tanto a las JSPs como a las interfaces de la capa de Negocio.

En el nivel de negocio, utilizaré los Services del Framework de Spring y sus interfaces para desacoplar la lógica de sus accesos.

Por otro lado, la parte de datos o capa de integración la implementaré usando Spring Data que mediante el Repository se realizarán las operaciones relacionadas con la BBDD (CRUD). A su vez utilizaré JPAs (Java Persistence Api) para la implementación de la persistencia de los datos. La base de datos que voy a utilizar va a ser una BBDD relacional y utilizaré PostgreSQL

2.10.1. Spring MVC

Este Framework proporciona una arquitectura basada en Modelo Vista Controlador, para el desarrollo de aplicaciones Web. Utiliza el patrón MVC que nos ayuda a separar las capas (Presentación, Negocio e Integración) manteniendo un acoplamiento entre ellos:

- El Modelo encapsula los datos de la aplicación
- La Vista se responsabiliza de mostrar al cliente la información utilizando HTML
- El Controlador se encarga de realizar la lógica de la aplicación solicitada por el usuario y le pasa dicha información a la vista para su representación gráfica.

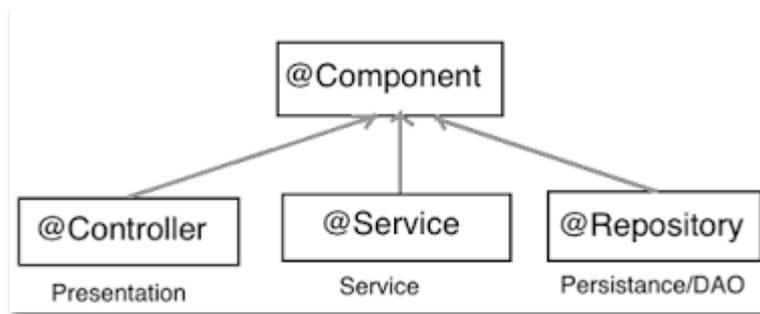


Ilustración 30 - Spring MVC

2.10.2. Otras tecnologías

- **Intelij IDEA:** IDE de desarrollo
- **Spring Boot:** Nos va a ayudar a centrarnos en el desarrollo de la aplicación evitando perder tiempo en configurar un Servidor Web así como en hacer los despliegues.
- **Maven:** Para ayudarnos con las librerías, dependencias y artefactos.
- **Spring Security:** Para ayudarnos con la seguridad (acceso de la aplicación).
- **MagicDraw:** Para la realización de los diagramas.
- **Pencil:** Para la realización del prototipo.

2.11. Análisis de Riesgos

Este proyecto, al estar desarrollada por una única persona, el riesgo más importante está directamente relacionada con el personal.

El TFG lo he dividido en varios entregables y en los dos primeros, el riesgo es de tiempo para desarrollar cada una de los contenidos. Sin embargo, para la fase de implementación, el riesgo mayor puede ser el desconocimiento de la tecnología que voy a utilizar en la implementación de la solución que estoy evaluando.

Otro de los riesgos es la indisposición de la persona que va a realizar todo el proyecto, que soy yo, que en el caso de que me surgiera algún contratiempo, no podría ser sustituida por nadie y eso ocasionaría un problema sin solución para este TFG. Ahora bien, en el análisis de riesgos por un lado tenemos los riesgos que pueden ser más importantes y por otro lado tenemos la probabilidad de que sucedan.

De los dos riesgos que estoy planteando, el primero que es la falta de conocimientos de las tecnologías que usaré, conllevaría el riesgo de poder retrasar el proyecto sustancialmente. En este caso, el riesgo es Medio y la probabilidad es Alta.

Para mitigar este riesgo, voy a utilizar una metodología agile, la cual me va a permitir realizar paquetes cerrados de desarrollos pequeños pero completos. Creo que de esta manera, podré ir avanzando poco a poco y podré ir aprendiendo a buen ritmo.

Con respecto al riesgo de tener un contratiempo, el riesgo es Alto, pero la probabilidad es Baja, porque aunque mis hijas pudieran darme algún susto, espero que se pueda solucionar en poco tiempo y me deje suficiente tiempo como para terminar el TFG.

En el caso de que este TFG fuese un proyecto real, el análisis de riesgos podría ser así:

<u>Concepto</u>	<u>Prob.(%)</u>	<u>Impacto</u>	<u>Estrategia mitigar riesgo</u>	<u>Riesgo</u>
Asociados al Proyecto				
Desviación ppto	30%	M	Incrementar un 30 % al costo real del software para mitigar las desviaciones.	0,60
Personal	14%	M	Motivación, formación a los stakeholder.	0,28
Recursos	10%	M	Optimizar los recursos.	0,20
Entendimiento con el Cliente	20%	A	Entablar una buena comunicación con el cliente.	0,60
Asociados a la Tecnología.				
Problemas de Diseño	5%	B	Realizar una checklist con las necesidades del cliente aceptando cada fase.	0,05
Problemas de Usabilidad	10%	B	Seguir criterios standard de usabilidad, y realizar Pruebas de usabilidad para comprobar el correcto funcionamiento del aplicativo.	0,10
Retraso por desconocimiento de las nuevas tecnologías	20%	A	Formación continua del personal.	0,60
Mantenimiento Evolutivo/Correctivo	5%	B	Definir claramente los conceptos que se incluyen en el trabajo para evitar posibles confusiones de conceptos.	0,05
Asociados al Negocio				
Riesgo Estratégico	10%	A	Realizar modificaciones requeridas por el cliente.	0,30
Riesgo de perder contacto con el cliente.	20%	M	Fijarlo por contrato el cobro de un % alto al inicio del proyecto. En caso de perderlo, ofrecerlo a otras	0,40

Ilustración 31 - Analisis de Riesgos

He definido tres niveles de impacto (A-Alto, M-Medio y B-Bajo) dependiendo del impacto que podría tener en el proyecto.

Teniendo en cuenta el impacto y la probabilidad, he ido ponderando un posible riesgo, teniendo los resultados de la tabla anterior.

También habría considerado los siguientes riesgos:

- Crítico** : Riesgos superiores al 0,70
- Alto** : Riesgos entre el 0,50 y el 0,70
- Medio** : Riesgos entre el 0,20 y el 0,50
- Bajo** : Riesgos inferiores al 0,20

Una vez realizado el análisis, haría un seguimiento especial para los riesgos mayores, para evitar que dichos riesgos se hicieran realidad y pudieran dañar el resultado final.

2.12. Prototipo gráfico inicial

He realizado un prototipo con la apariencia inicial y la movilidad entre pantallas. El resultado es el siguiente:

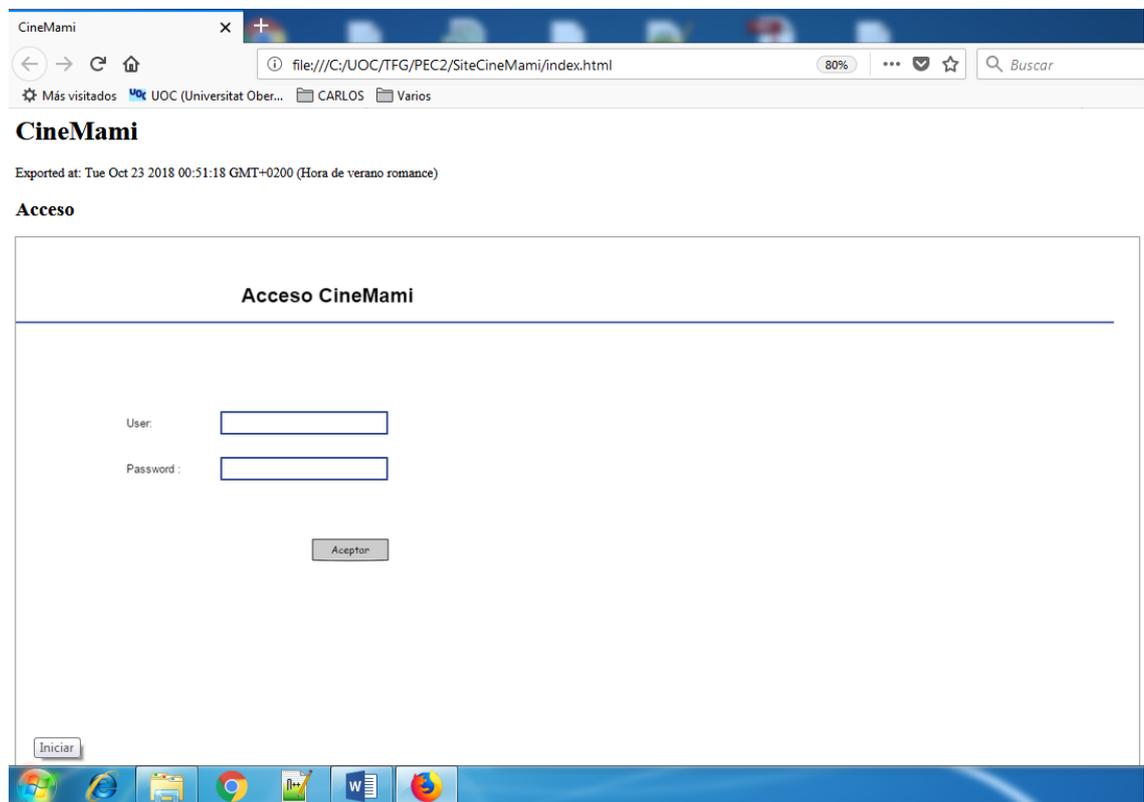


Ilustración 32 - Prototipo Acceso

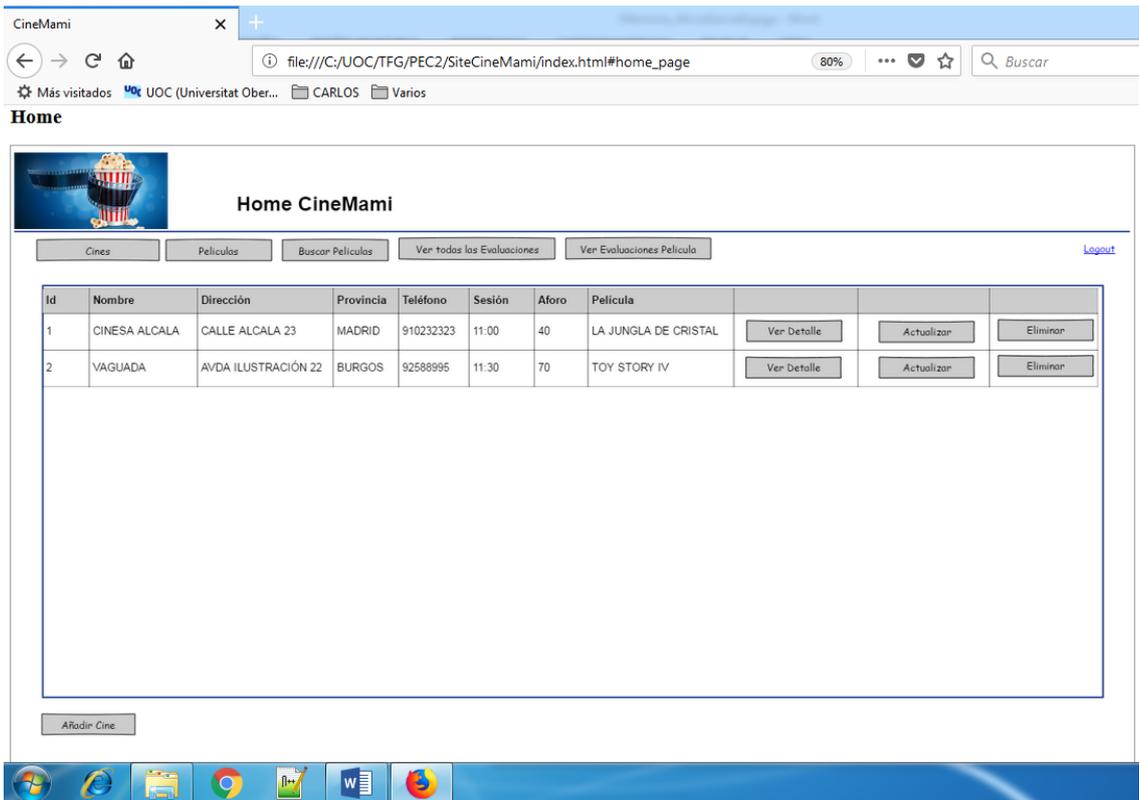


Ilustración 33 - Prototipo Cines

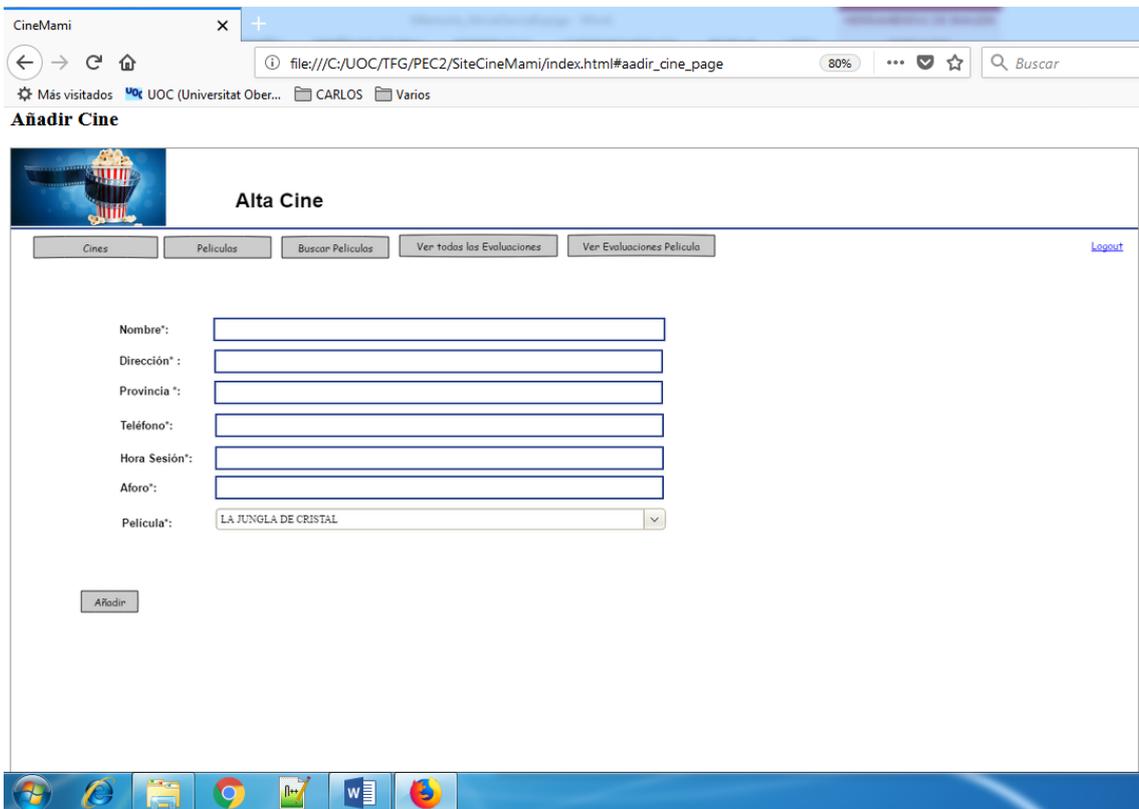


Ilustración 34 - Prototipo Alta Cine

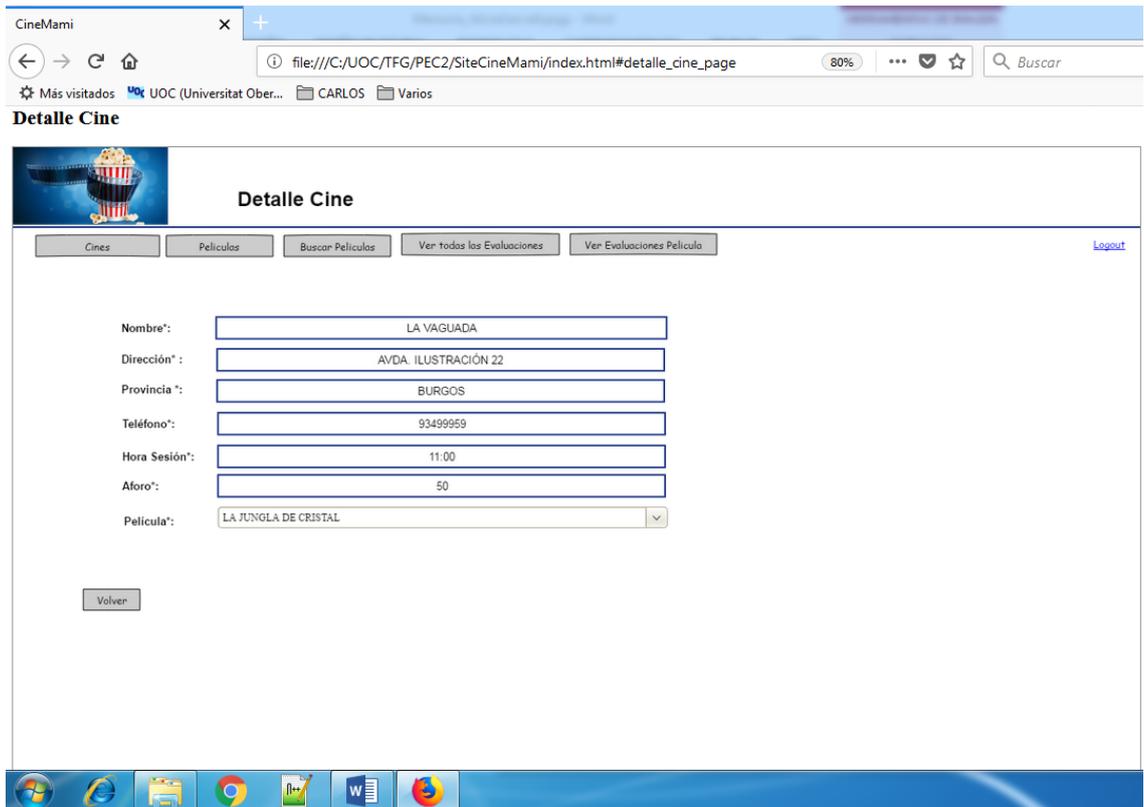


Ilustración 35 - Prototipo Detalle Cine

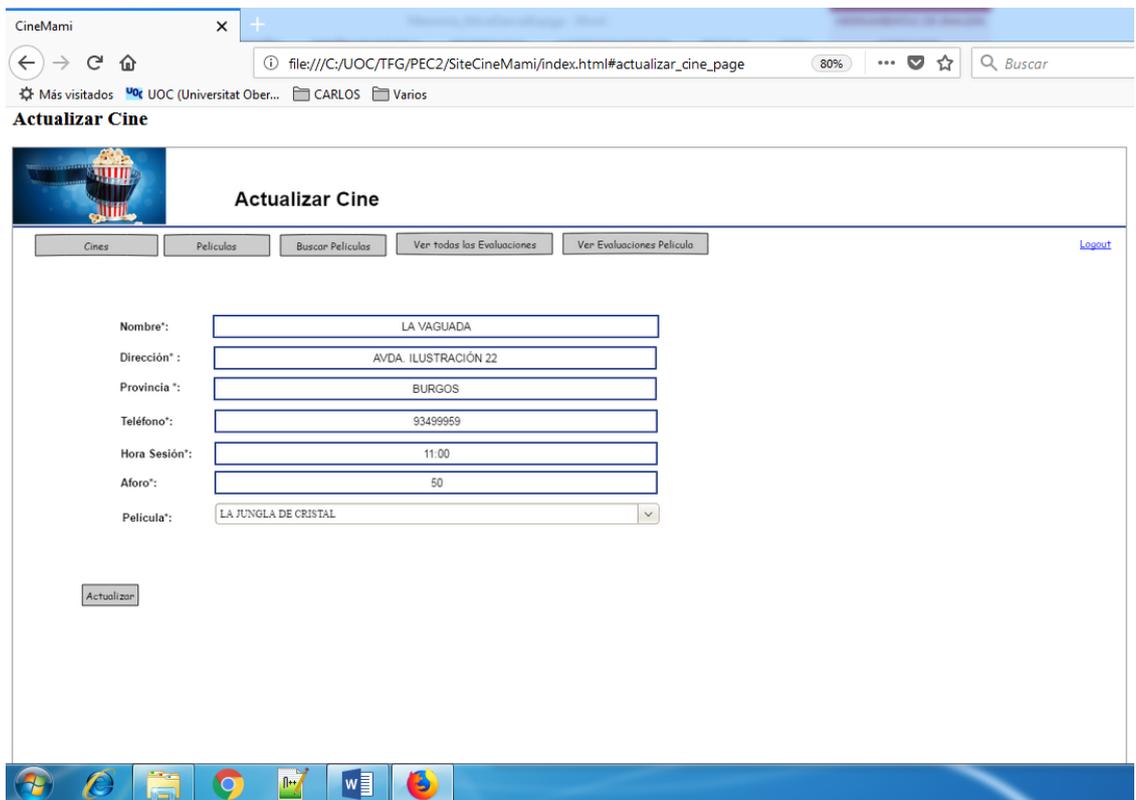


Ilustración 36 - Prototipo Actualizar Cine

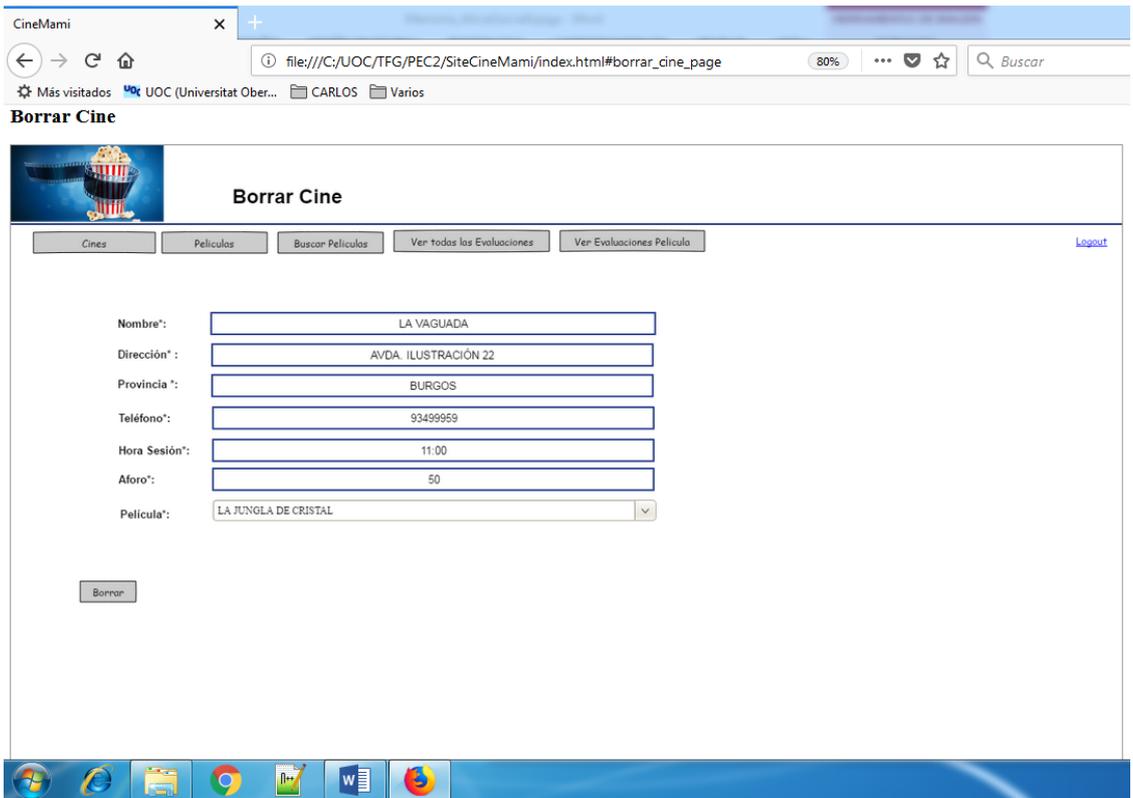


Ilustración 37 - Prototipo Borrar Cine

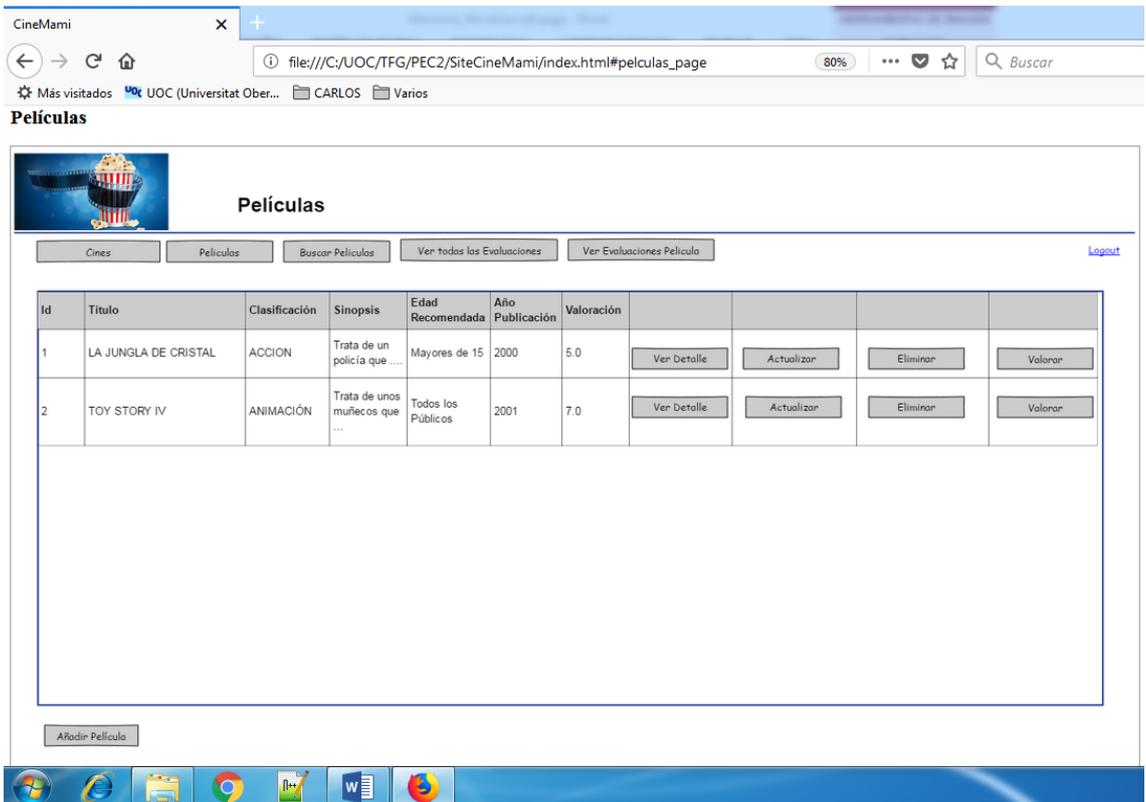


Ilustración 38 - Prototipo Películas

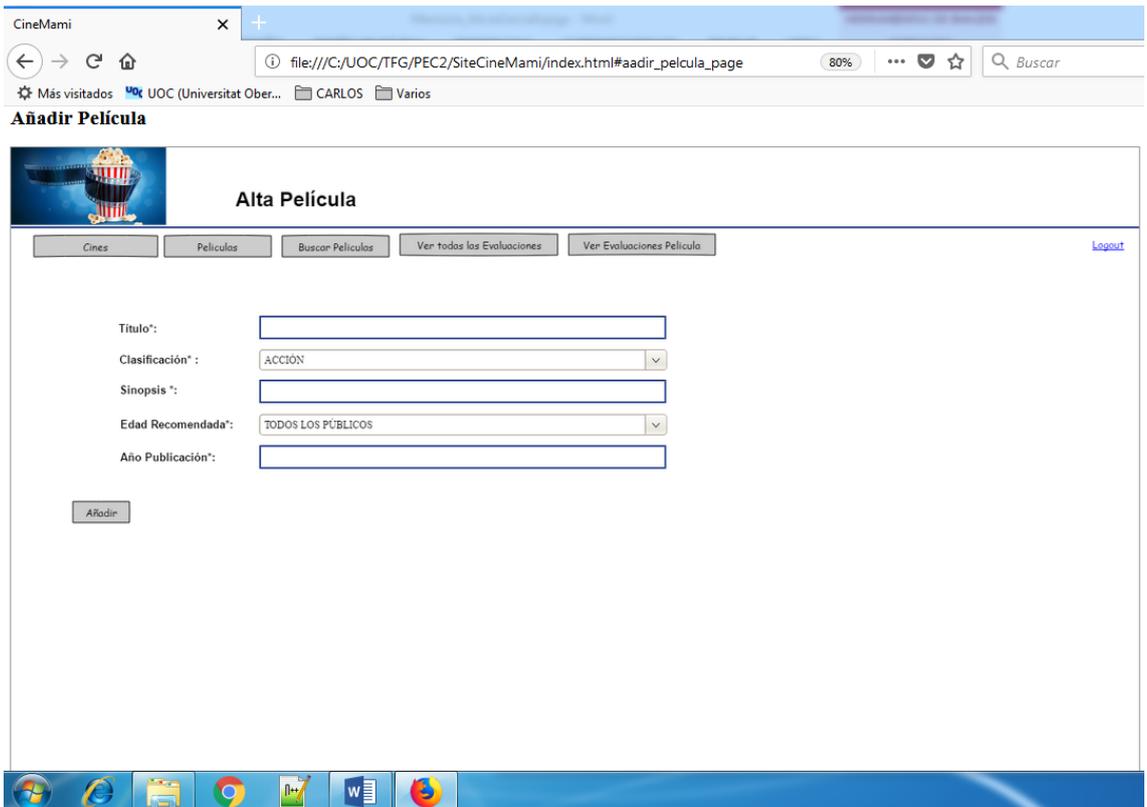


Ilustración 39 - Prototipo Alta Película

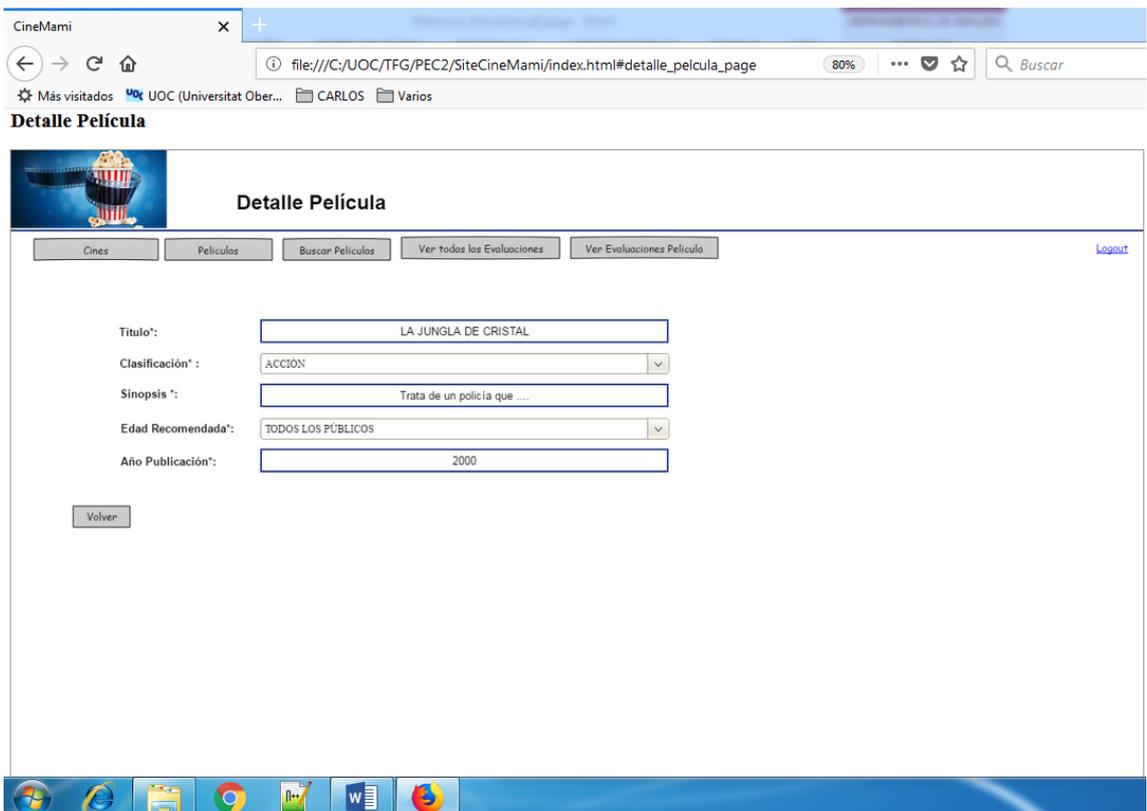


Ilustración 40 - Prototipo Detalle Película

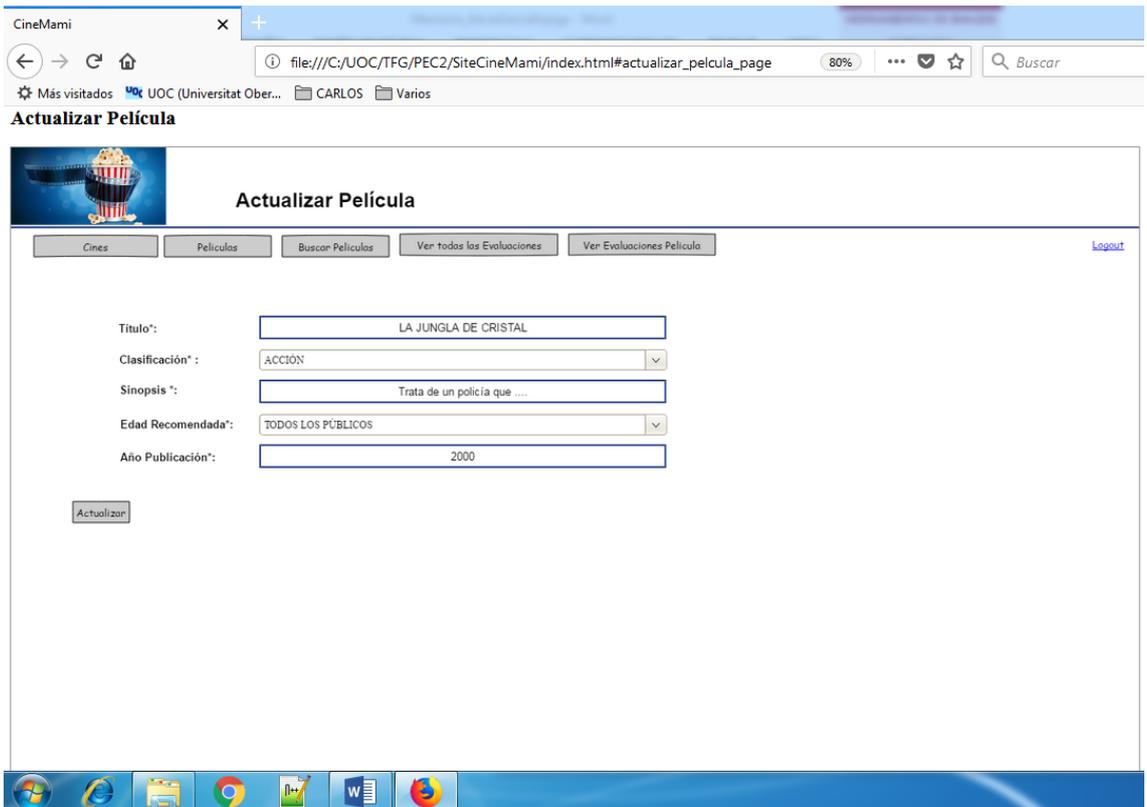


Ilustración 41 - Prototipo Actualizar Película

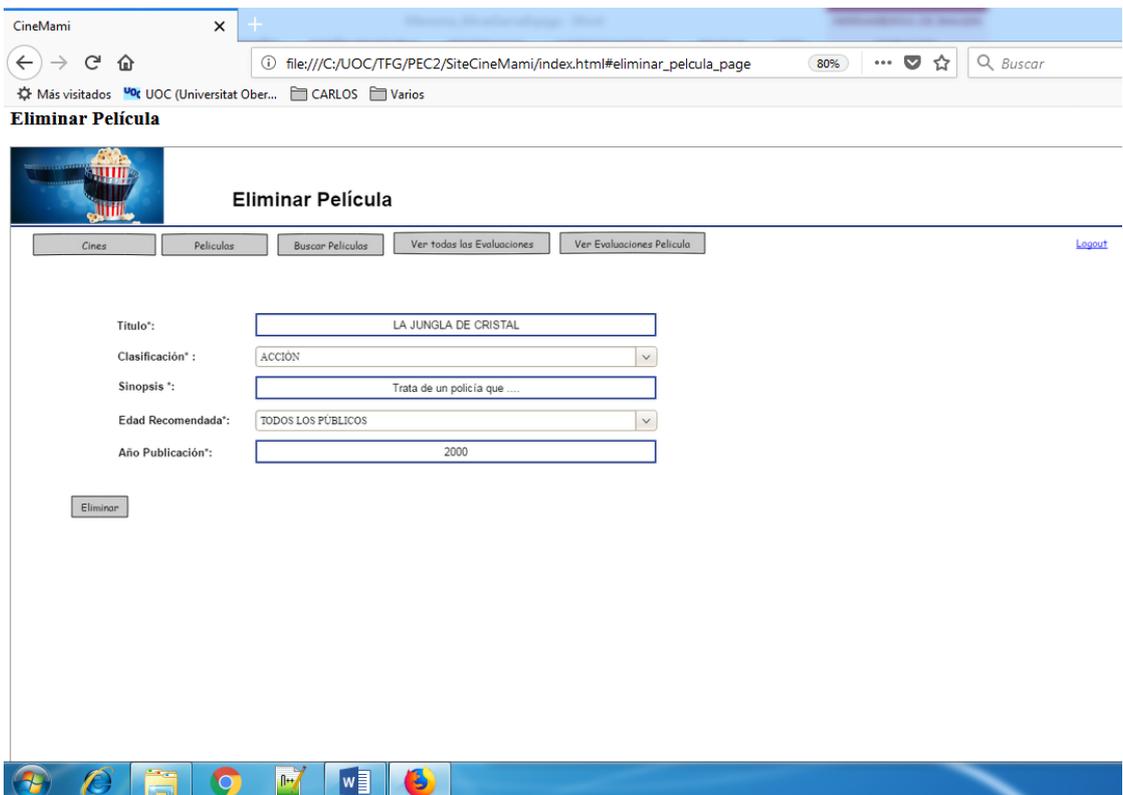


Ilustración 42 - Prototipo Eliminar Película

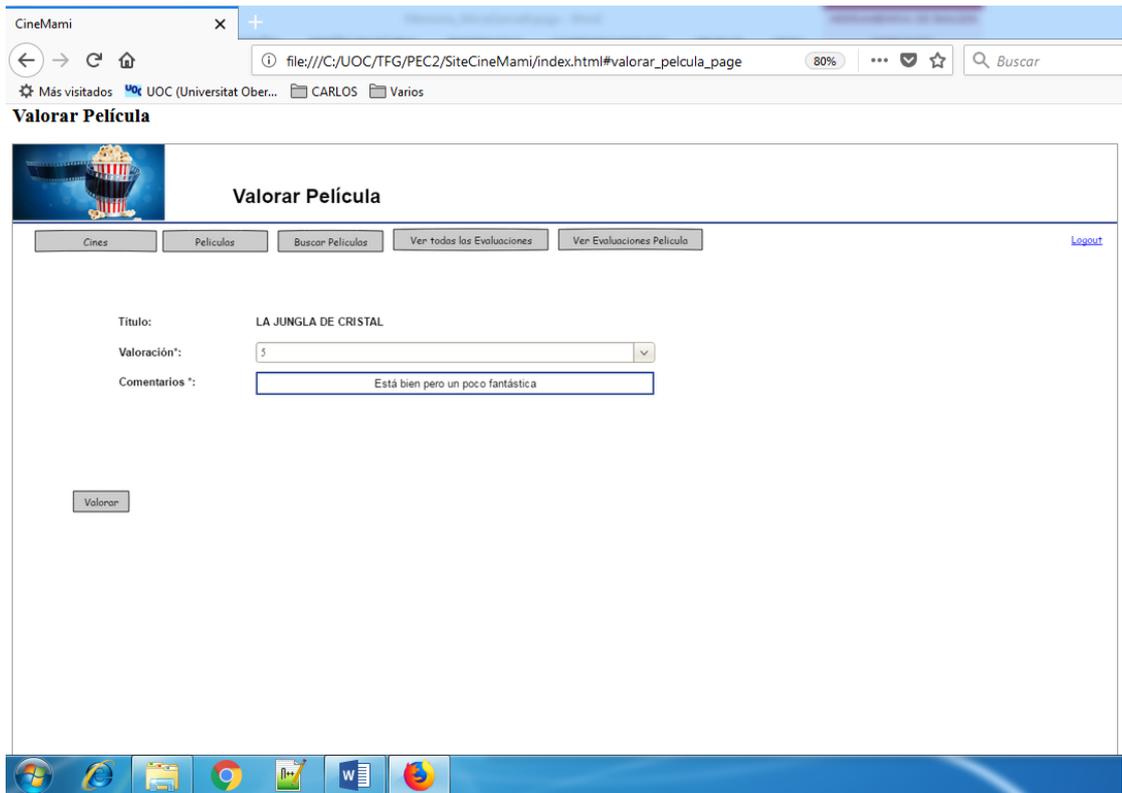


Ilustración 43 - Prototipo Valorar Película

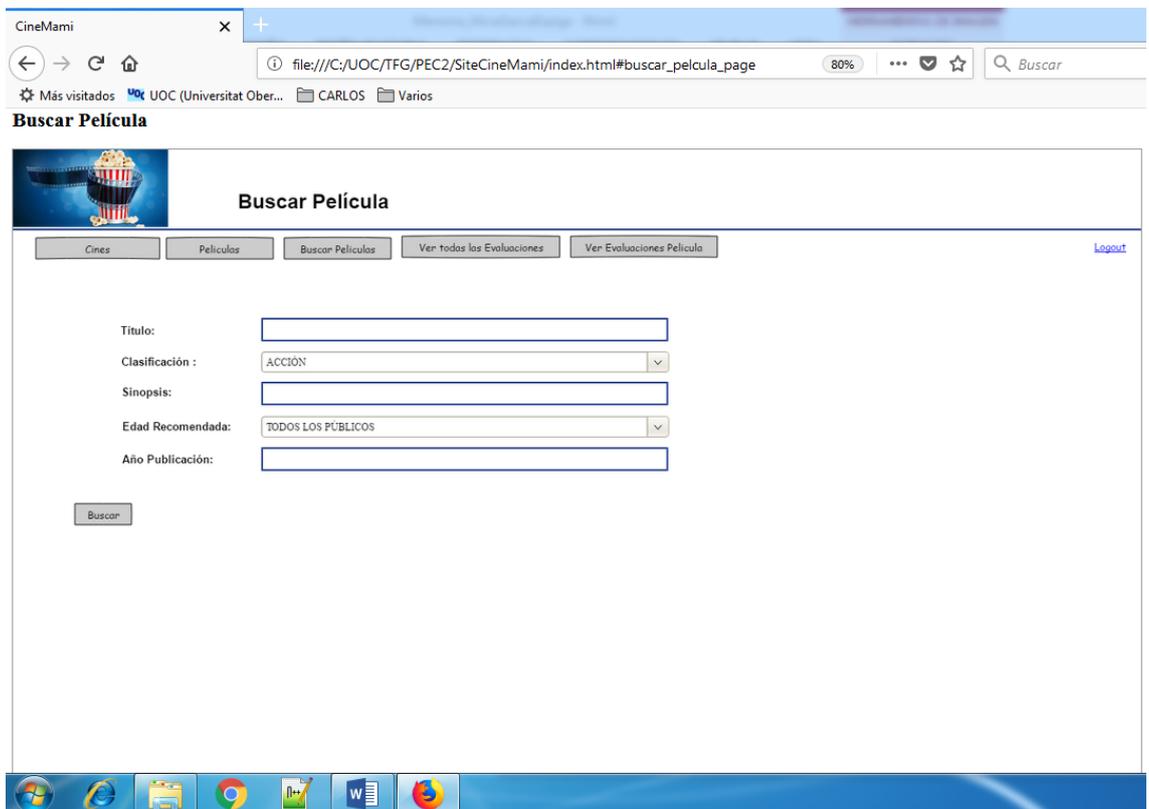


Ilustración 44 - Prototipo Buscar Película

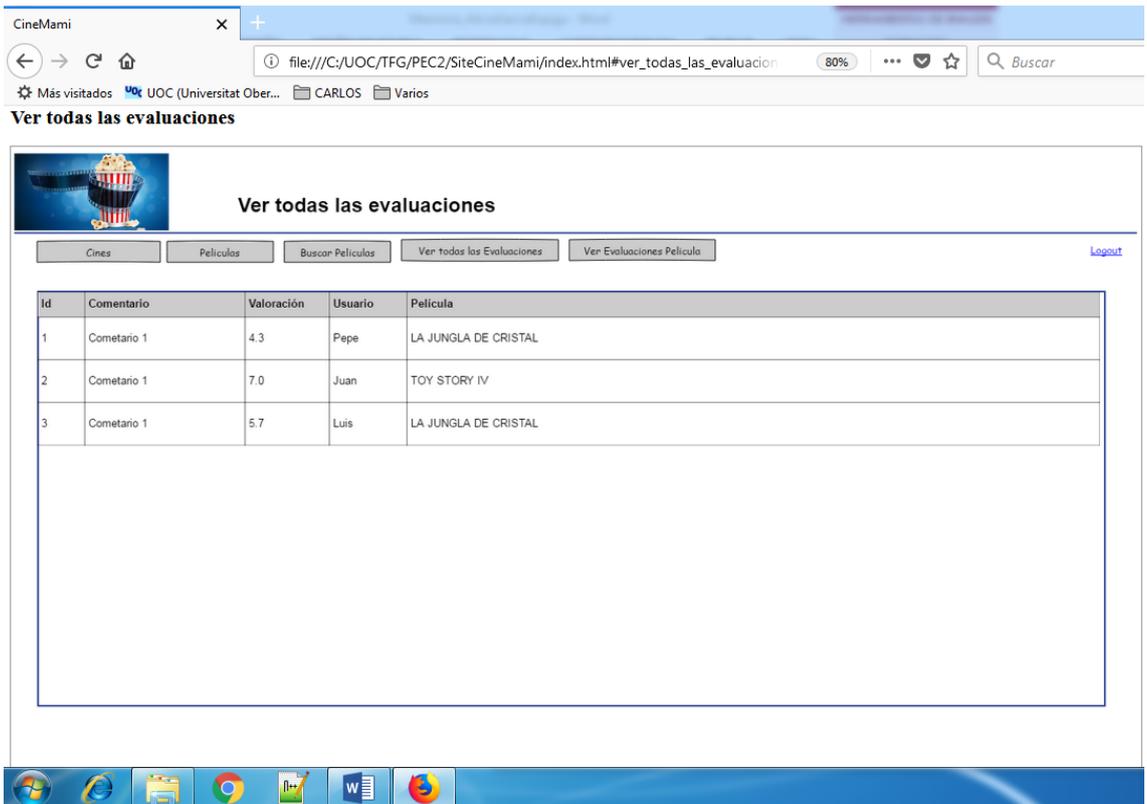


Ilustración 45 - Prototipo Ver todas las Evaluaciones

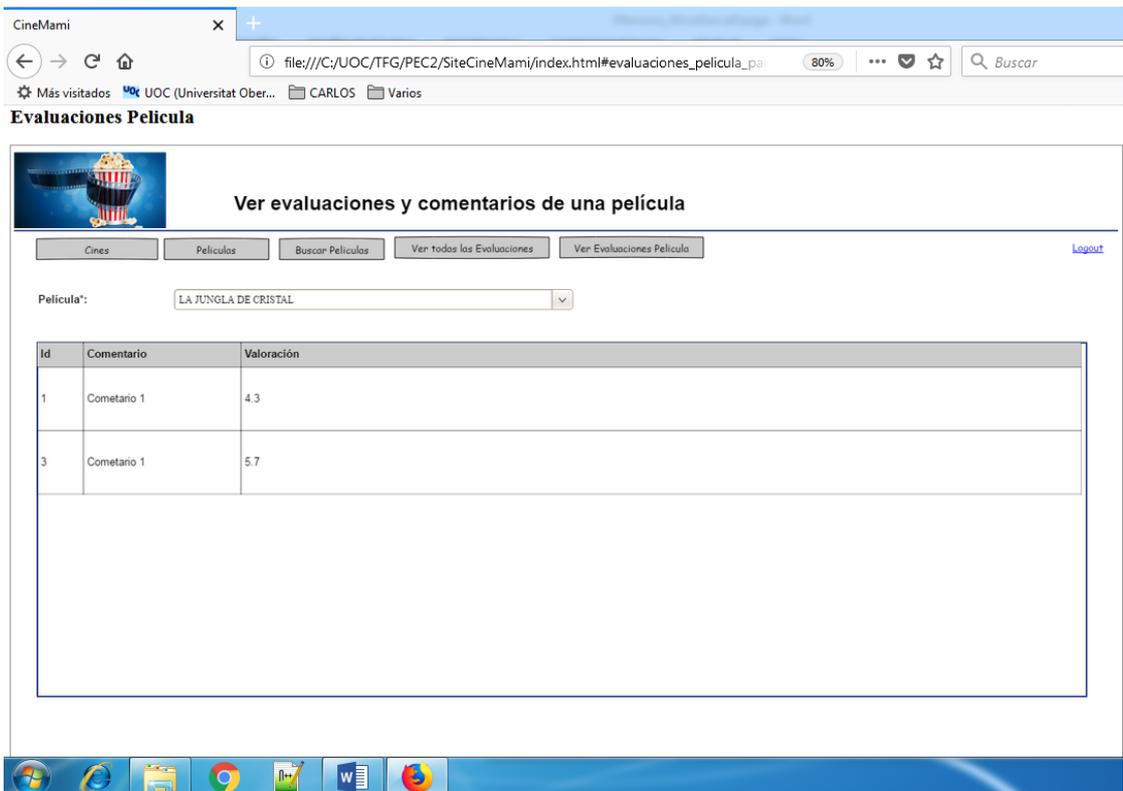


Ilustración 46 - Prototipo Ver evaluaciones y comentarios de una película

3. Implementación y programación

3.1. Experiencia en el desarrollo del proyecto:

Como resumen de la experiencia podría decir que ha sido muy gratificante tanto en lo personal como en lo académico. Poder realizar un desarrollo con tecnologías de última generación y aprender a usar estas tecnologías ha sido un reto difícil, pero muy satisfactorio.

Al principio, me entró la duda de realizar la implementación de esta solución utilizando una tecnología conocida, tipo .Net, pero de haberlo realizado así, habría sido un desarrollo más y no me habría aportado nada nuevo. Ahora me alegro de haber elegido una tecnología que era desconocida para mí y de la cual ahora conozco bastante.

He tenido que luchar con dos grandes retos. Uno ha sido el que he comentado que es el uso de una tecnología que sólo tenía conocimientos básicos y el otro gran reto ha sido poder compaginar mi vida laboral, con la personal y con el desarrollo de este proyecto. He tenido que ponerme con el TFG cuando acostaba a mis pequeñas (Leire de 4 años y Paula de 2) y os aseguro que las ganas que quedan después de un día intenso a las 22.00h son escasas. Pero siempre he sacado fuerzas, no sé realmente de donde, para ir avanzando poco a poco hasta llegar a la meta.

Durante la investigación de la tecnología, he encontrado dos puntos muy dispares. El hecho de tener mucha información es muy bueno para ver cómo se pueden hacer lo mejor posible los desarrollos, pero también es un problema el disponer de tanta información, porque llega un momento en el que no sabes bien que información es la veraz. También enfocado a uno de mis grandes retos, el hecho de tener mucha información, te obliga a gastar mucho tiempo en estudiar todo lo que crees que puede ser interesante para tu proyecto, y en mi caso he tenido que restar horas de sueño para poder llevarlo a cabo.

Lo primero que hice para comenzar el proyecto, una vez instalado todo el software y las utilidades comentadas anteriormente en este documento, fue montar la Base de Datos. Creé a mano la base de datos en Postgresql, pero posteriormente me di cuenta de que no había sido necesaria esa tarea. Utilizando las JPA's de Spring Data, te hace esta operativa automáticamente, pero eso lo descubrí un poco más tarde.

Una vez tenía el modelo de datos, me puse a realizar una primera prueba con un formulario sin nada de diseño, las operaciones básicas CRUD. Al principio me fue funcionando, pero cuando le metí las relaciones, empezaron algunos problemas. Estuve buscando información tanto en el libro de Spring 3 como en internet y descubrí que me faltaba configurar uno de los parámetros de las relaciones en las JPA's. En concreto el orphanRemoval.

Una vez he conseguido que funcionasen las operaciones básicas, me puse a crear las búsquedas aplicando filtros y la verdad es que di también con la solución bastante rápido. Me ha gustado mucho como ayuda el Framework de Spring, en este sentido también.

Estuve realizando pruebas por cada uno de los casos de uso y una vez eran satisfactorias, pasaba al siguiente caso hasta finalizar todos.

He añadido en cada JPA el atributo "Version" para evitar el problema de concurrencia. En este sentido con el Tag @Version, también se evita la concurrencia automáticamente. He realizado diferentes pruebas y he comprobado cómo va cambiando este valor en BBDD según se van realizando operaciones.

Para ir analizando paso a paso cada pieza de desarrollo, implementé un logger para ir viendo en consola el resultado de cada variable en cada paso por el programa.

El siguiente paso fue dar un poco de diseño al proyecto. Para ello me apoyé en Bootstrap. Esta parte me ha llevado tiempo, porque había muchas formas de hacer cada cosa y hasta que lo he visto más o menos bonito, le tuve que dedicar un poco de tiempo en modo prueba y error.

La parte de Spring Security en la funcionalidad básica me funcionó bien, pero intenté realizar un formulario personalizado de login y no he tenido tiempo de llegar a implementarlo.

3.2. Mejoras para siguientes versiones:

De las mejoras que voy a sugerir, creo que las más importantes son las relativas a la Seguridad. Por ejemplo, las password deberían estar cifradas y aplicar algún algoritmo de cifra para hacer las comprobaciones oportunas. Otro punto también relativo a la seguridad, es que he intentado implementar un formulario personalizado, pero por falta de tiempo no fui capaz de dejarlo funcionando. Creo que es importante esta mejora, para evitar que los hacker puedan tener pistas acerca de cómo se ha desarrollado esta aplicación.

Creo que para las Redes Sociales estaría bien poder automatizar las publicaciones. Se podría acceder al Sistema usando un perfil de la red social en cuestión y que los administradores pudieran hacer publicaciones.

Crear un nuevo Rol que sea para el Cine. Se podría crear un nuevo perfil del tipo CINE para que los propios Cines se pudieran conectar y ver los comentarios de los usuarios con respecto a estas sesiones. En esta línea se debería implementar un servicio tipo Chat para que pudieran intercambiar mensajes los Cines con los usuarios.

Un usuario no debería poder editar una película.

3.3. Resultado role Admin:

Login with Username and Password

User:

Password:

Credenciales : (User: ADMIN Password : ADMIN)

Home:

 Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Id	Nombre	Dirección	Provincia	Teléfono	Sesión	Aforo	Película			
1	CINE ALCALA	CALLE ALCALA	MADRID	9191919191	11:00	45	LA JUNGLA DE CRISTAL	<input type="button" value="Detalle"/>	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>
2	CINE LA VAGUADA	AVDA LA ILUSTRACION	MADRID	912222222	12:00	70	LA JUNGLA DE CRISTAL	<input type="button" value="Detalle"/>	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>
4	CINE GRAN VIA	GRAN VIA	MADRID	9122333344	11:00	80	TOY STORY	<input type="button" value="Detalle"/>	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>
6	CINES JAEN	C/ CUENCA 24	JAEN	934555967	10:45	70	GHOST	<input type="button" value="Detalle"/>	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>

[Añadir Cine](#)

Ven a Cine con tu bebe - CineMami

Alta Cine:

 Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Nombre

Dirección

Provincia

Teléfono

Hora Sesión

Aforo

Película:

Ven a Cine con tu bebe - CineMami

Detalle Cine:

 Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Nombre
CINE ALCALA

Dirección
CALLE ALCALA

Provincia
MADRID

Teléfono
9191919191

Hora Sesión
11:00

Aforo Sala
45

Pelicula
LA JUNGLA DE CRISTAL

[Volver](#)

Editar Cine:

 Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Nombre
CINE ALCALA

Dirección
CALLE ALCALA

Provincia
MADRID

Teléfono
9191919191

Hora Sesión
11:00

Aforo
45

Pelicula: LA JUNGLA DE CRISTAL ▼

[Actualizar](#)

Ven a Cine con tu bebe - CineMami

Borrar Cine:



Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Nombre

CINE ALCALA

Dirección

CALLE ALCALA

Provincia

MADRID

Teléfono

9191919191

Hora Sesión

11:00

Aforo Sala

45

Película

LA JUNGLA DE CRISTAL

Borrar

Películas:



Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Id	Título	Clasificación	Sinopsis	Edad recomendada	Publicación	Valoración				
1	GHOST	AMOR	Un hombre es asesinado y su fantasma ayuda a su novia	TP	1990	5.0	Detalle	Editar	Borrar	Valorar
2	TOY STORY	ANIMACION	Historia de unos juguetes	TRES	2001	4.0	Detalle	Editar	Borrar	Valorar
3	LA JUNGLA DE CRISTAL	ACCION	Un agente tiene que salvar el mundo	QUINCE	2000	6.3333335	Detalle	Editar	Borrar	Valorar
6	ANT-MAN	ANIMACION	Un hombre se somete a una prueba.	TP	2008	8.0	Detalle	Editar	Borrar	Valorar

[Añadir Película](#)

Ven a Cine con tu bebe - CineMami

Añadir Película:



Cine Mami - Cines **Peliculas** [Buscar Peliculas](#) [Ver todas las Evaluaciones](#) [Ver Evaluaciones Película](#) [Usuario : ADMIN](#) [Logout](#)

Título

Clasificación: *** Clasificacion *** ▼

Síntesis

Edad recomendada: *** Edad Recomendada *** ▼

Año Publicación

Añadir

Ven a Cine con tu bebe - CineMami

Detalle Película:



Cine Mami - Cines **Peliculas** [Buscar Peliculas](#) [Ver todas las Evaluaciones](#) [Ver Evaluaciones Película](#) [Usuario : ADMIN](#) [Logout](#)

Título

Clasificación

Síntesis

Edad recomendada

Año Publicación

Volver

Ven a Cine con tu bebe - CineMami

Editar Película:



Cine Mami - Cines **Peliculas** [Buscar Peliculas](#) [Ver todas las Evaluaciones](#) [Ver Evaluaciones Película](#) [Usuario : ADMIN](#) [Logout](#)

Título

Clasificación: AMOR ▼

Síntesis

Edad recomendada: TP ▼

Año Publicación

Actualizar

Ven a Cine con tu bebe - CineMami

Borrar Película:



Cine Mami - Cines **Películas** Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película [Usuario : ADMIN](#) [Logout](#)

Título

GHOST

Clasificación

AMOR

Síntesis

Un hombre es asesinado y su fantasma ayuda a su novia

Edad recomendada

TP

Año Publicación

1990

Borrar

Ven a Cine con tu bebe - CineMami

Valorar Película:



Cine Mami - Cines **Películas** Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película [Usuario : ADMIN](#) [Logout](#)

Evaluación

Valoración y comentarios para la película

Película: **TOY STORY**

Comentario

comentario

Valoración

4

Grabar

Ven a Cine con tu bebe - CineMami

Buscar Película:

Cine Mami - Cines Películas **Buscar Películas** Ver todas las Evaluaciones Ver Evaluaciones Película [Usuario : ADMIN](#) Logout

Título

Clasificación: ***** Clasificacion ***** ▼

Sinopsis

Edad recomendada: ***** Edad Recomendada ***** ▼

Año Publicación

Ven a Cine con tu bebe - CineMami

Resultado Búsqueda:

Cine Mami - Cines Películas **Buscar Películas** Ver todas las Evaluaciones Ver Evaluaciones Película [Usuario : ADMIN](#) Logout

Id	Título	Clasificación	Sinopsis	Edad recomendada	Publicación	Valoración				
6	ANT-MAN	ANIMACION	Un hombre se somete a una prueba.	TP	2008	8.0	<input type="button" value="Detalle"/>	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>	<input type="button" value="Valorar"/>
1	GHOST	AMOR	Un hombre es asesinado y su fantasma ayuda a su novia	TP	1990	5.0	<input type="button" value="Detalle"/>	<input type="button" value="Editar"/>	<input type="button" value="Borrar"/>	<input type="button" value="Valorar"/>

[Añadir Película](#)

Ven a Cine con tu bebe - CineMami

Ver todas las Evaluaciones:

Cine Mami - Cines Películas **Ver todas las Evaluaciones** Ver Evaluaciones Película [Usuario : ADMIN](#) Logout

Id	Comentario	Valoracion	Usuario	Pelicula	
1	Comentario 1	2	ADMIN	GHOST	<input type="button" value="Delete"/>
3	COMENTARIO USUARIO 1	8	USER1	GHOST	<input type="button" value="Delete"/>
5	comentario	4	ADMIN	TOY STORY	<input type="button" value="Delete"/>
9	soy un admin y me ha gustado	8	ADMIN	LA JUNGLA DE CRISTAL	<input type="button" value="Delete"/>
10	SOY USER1 y valoro	4	USER1	LA JUNGLA DE CRISTAL	<input type="button" value="Delete"/>
11	SOY USER2 Y VALORO	7	USER2	LA JUNGLA DE CRISTAL	<input type="button" value="Delete"/>
13	ESTA BIEN. ANIMADA	8	USER2	ANT-MAN	<input type="button" value="Delete"/>

Ven a Cine con tu bebe - CineMami

Borrar valoración y comentarios:

Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Id
10

Comentario
SOY USER1 y valoro

Valoración
4

Usuario
USER1

Película
LA JUNGLA DE CRISTAL

Delete

Ver Evaluaciones y comentarios de una película:

Cine Mami - Cines Películas Buscar Películas Ver todas las Evaluaciones Ver Evaluaciones Película Usuario : ADMIN Logout

Comentarios película

Película : LA JUNGLA DE CRISTAL Evaluacion : 6,33

Seleccionar

Id	Comentario	Valoracion
9	soy un admin y me ha gustado	8
10	SOY USER1 y valoro	4
11	SOY USER2 Y VALORO	7

Evaluaciones de la película seleccionada

Ven a Cine con tu bebe - CineMami

Al pulsar en LOGOUT vuelve a la pantalla de LOGIN

3.4. Resultado role USER:

Login with Username and Password

User:

Password:

Login

Credenciales : (User: USER1 Password : USER1)

HOME:



Cine Mami - Cines Peliculas Buscar Peliculas Usuario : USER1 Logout

Id	Nombre	Dirección	Provincia	Teléfono	Sesión	Aforo	Película
1	CINE ALCALA	CALLE ALCALA	MADRID	9191919191	11:00	45	LA JUNGLA DE CRISTAL
2	CINE LA VAGUADA	AVDA LA ILUSTRACION	MADRID	9122222222	12:00	70	LA JUNGLA DE CRISTAL
4	CINE GRAN VIA	GRAN VIA	MADRID	9122333344	11:00	80	TOY STORY
6	CINES JAEN	C/CUENCA 24	JAEN	934555967	10:45	70	GHOST

Ven a Cine con tu bebe - CineMami

Películas:



Cine Mami - Cines Peliculas Buscar Peliculas Usuario : USER1 Logout

Id	Título	Clasificación	Sinopsis	Edad recomendada	Publicación	Valoración			
1	GHOST	AMOR	Un hombre es asesinado y su fantasma ayuda a su novia	TP	1990	5.0	Detalle	Editar	Valorar
2	TOY STORY	ANIMACION	Historia de unos juguetes	TRES	2001	4.0	Detalle	Editar	Valorar
3	LA JUNGLA DE CRISTAL	ACCION	Un agente tiene que salvar el mundo	QUINCE	2000	6.3333335	Detalle	Editar	Valorar
6	ANT-MAN	ANIMACION	Un hombre se somete a una prueba.	TP	2008	8.0	Detalle	Editar	Valorar

Ven a Cine con tu bebe - CineMami

Detalle Película:



Cine Mami - Cines Peliculas Buscar Peliculas Usuario : USER1 Logout

Título

GHOST

Clasificación

AMOR

Sinopsis

Un hombre es asesinado y su fantasma ayuda a su novia

Edad recomendada

TP

Año Publicación

1990

[Volver](#)

Ven a Cine con tu bebe - CineMami

Editar Película:



Cine Mami - Cines Películas Buscar Películas Usuario : USER1 Logout

Título

Clasificación: AMOR ▼

Sinopsis

Edad recomendada: TP ▼

Año Publicación

Actualizar

Ven a Cine con tu bebe - CineMami

Valorar Película:



Cine Mami - Cines Películas Buscar Películas Usuario : USER1 Logout

Evaluación

Valoración y comentarios para la película

Película: LA JUNGLA DE CRISTAL

Comentario

Valoración

Grabar

Ven a Cine con tu bebe - CineMami

Buscar Película:



Cine Mami - Cines Películas Buscar Películas Usuario : USER1 Logout

Título

Clasificación: *** Clasificacion *** ▼

Sinopsis

Edad recomendada: *** Edad Recomendada *** ▼

Año Publicación

Buscar

Resultado Búsqueda:



Cine Mami - Cines [Películas](#) [Buscar Películas](#) [Usuario : USER1](#) [Logout](#)

Id	Título	Clasificación	Sinopsis	Edad recomendada	Publicación	Valoración			
3	LA JUNGLA DE CRISTAL	ACCION	Un agente tiene que salvar el mundo	QUINCE	2000	6.3333335	Detalle	Editar	Valorar

[Ven a Cine con tu bebe - CineMami](#)

4. Pruebas

4.1. Resumen de las pruebas realizadas por sistema:

<u>SISTEMA</u>	<u>RESULTADO</u>
Administración del sistema	
Identificarse en el sistema	Prueba OK
Salir del sistema	Prueba OK
Administración Cine	
Dar de alta un Cine	Prueba OK
Dar de baja un Cine	Prueba OK
Modificar un Cine	Prueba OK
Ver el detalle de un Cine	Prueba OK
Administración Película	
Dar de alta una Película	Prueba OK
Dar de baja una Película	Prueba OK
Modificar una Película	Prueba OK
Evaluar una Película	Prueba OK
Ver el detalle de una Película	Prueba OK
Buscar una Película aplicando filtros	Prueba OK
Ve evaluación de una Película	Prueba OK
Administración Evaluación	
Ver comentarios y evaluaciones de una Película	Prueba OK
Eliminar comentarios y evaluaciones de una Película	Prueba OK
Ver todas las evaluaciones	Prueba OK
Seguridad	
Control de acceso	Prueba OK
Control de permisos	Prueba OK
Formulario de acceso Propio	Prueba KO

4.2. Detalle de las pruebas realizadas por componente y caso de uso:

Siguiendo los casos de uso este ha sido el resultado de las prueba de cada uno de los componentes. Si este proyecto fuese realizado en equipo, sería recomendable que cada componente fuese probado por personas diferentes a las que lo desarrollaron, pero en este caso que es mi trabajo de fin de grado, todos han sido probados por mí.

Componente	Administración del sistema		
Caso	Login		
1	Acceso con credenciales correctas	Acceder	OK
2	Acceso con credenciales incorrectas	Dar error y no permitir acceso	OK
3	Acceso diferentes roles	Acceder y visualizar lo que puede ver	OK
4	Utilizar un formulario personalizado de login	Acceder	KO

Componente	Administración del sistema		
Caso	Logout		
1	Salir del Sistema	Cerrar sesión y volver a la pantalla de login	OK
2	Acceder directamente a una url con permisos de un usuario una vez realizado	No permitir acceso	OK

Componente	Administración Cine		
Caso	Alta		
1	Pulsar Alta Cine	Mostrar Formulario con los campos del alta	OK
2	Guardar Alta Cine	Guardar datos en BBDD	OK
3	Introducir datos erróneos	Devuelve error	OK

Componente	Administración Cine		
Caso	Baja		
1	Pulsar Baja Cine	Mostrar Formulario con los campos para confirmar baja	OK
2	Borrar Cine	Borrar datos en BBDD	OK

Componente	Administración Cine		
Caso	Modificar		
1	Pulsar Modificar Cine	Mostrar Formulario con los campos para modificar	OK
2	Guardar modificaciones	Modificar datos en BBDD	OK

Componente	Administración Cine		
Caso	Ver Detalle		
1	Pulsar Detalle Cine	Mostrar Formulario con los datos del cine	OK
2	Intentar actualizar datos del Cine	No lo permite	OK

Componente	Administración Película		
Caso	Alta		
1	Pulsar Alta Película	Mostrar Formulario con los campos del alta	OK
2	Guardar Alta Película	Guardar datos en BBDD	OK
3	Introducir datos erróneos	Devuelve error	OK

Componente	Administración Película		
Caso	Baja		
1	Pulsar Baja Película	Mostrar Formulario con los campos para confirmar baja	OK
2	Borrar Película	Borrar datos en BBDD	OK

Componente	Administración Película		
Caso	Modificar		
1	Pulsar Modificar Película	Mostrar Formulario con los campos para modificar	OK
2	Guardar modificaciones	Modificar datos en BBDD	OK
3	Guardar modificaciones con datos erróneos	Mostrar error	OK

Componente	Administración Película		
Caso	Valorar Película		
1	Pulsar Valorar Película	Mostrar Formulario con los datos para valorar	OK
2	Pulsar Guardar Valoración	Guardar valoración en BBDD	OK
3	Pulsar Guardar Valoración con datos erróneos	Mostrar error	OK
4	Intentar valorar una película valorada anteriormente	Muestra valoración anterior y le deja modificarla	OK

Componente	Administración Película		
Caso	Ver Detalle		
1	Pulsar Detalle Película	Mostrar Formulario con los datos de la película	OK
2	Intentar actualizar datos	No lo permite	OK

Componente	Administración Película		
Caso	Buscar Película aplicando filtros		
1	Pulsar Buscar Película	Mostrar Formulario para aplicar filtro	OK

2	Introducir datos y pulsar en Buscar	Mostrar resultados	OK
---	-------------------------------------	--------------------	----

Componente	Administración Evaluación		
Caso	Ver comentarios y evaluaciones de una Película		
1	Seleccionar película y pulsar en ver	Mostrar Formulario con los datos de las valoraciones	OK
2	Seleccionar película sin valoraciones	Muestra una lista vacía	OK

Componente	Administración Evaluación		
Caso	Eliminar comentarios y evaluaciones de una Película		
1	Ver todas las evaluaciones y pulsar en Delete	Mostrar Formulario con los datos de las valoraciones a eliminar	OK
2	Confirmar pulsando Delete	Borrar datos de la BBDD	OK

Componente	Administración Evaluación		
Caso	Ver todas las evaluaciones		
1	Pulsar en ver todas las evaluaciones	Mostrar Formulario con los datos de todas las valoraciones	OK
2	Confirmar pulsando Delete	Borrar datos de la BBDD	OK

5. Conclusiones

La realización de mi TFG ha sido un gran reto para mí. He conseguido aplicar el conocimiento adquirido durante toda la carrera en el uso de nuevas tecnologías para realizar este proyecto. Aunque me considero una programadora con experiencia, el reto de utilizar tecnologías desconocidas para mí, has sido un gran reto que he tenido que ir superando poco a poco.

No ha sido el único reto. Compaginar el proyecto con mi vida personal ha sido también otro reto importante. Ha sido mucho el esfuerzo realizado de muchas noches sin apenas dormir, teniendo que esperar para acostar a mis hijas para ponerme con el proyecto, pero al final ha merecido la pena.

Mi carrera profesional ha estado más orientada al desarrollo de aplicaciones cliente-servidor en .net. Creo que me han aportado tranquilidad en las fases de análisis y diseño, pero poco me ha aportado a la hora de llevar a cabo la implementación. Aunque he ido cumpliendo hitos y tiempos que había plasmado en el planning inicial, bien es cierto que en determinados momentos he sentido que la falta de conocimiento en el entorno me estaba llevando a situaciones incómodas, pero gracias a mis ganas de aprender, a mi dedicación y persistencia he conseguido cumplir los objetivos marcados para esta asignatura.

Mis conocimientos en la tecnología elegida para la realización del TFG eran mínimos. Quizá haya sido el mayor problema que me he encontrado para cumplir con el programa marcado. Cuando me encontraba en un callejón sin salida, volvía a replantear el problema para busca de nuevas soluciones. Gracias a ir avanzando poco a poco conseguí coger fuerzas para continuar con el proyecto hasta su fin. Creo que puedo garantizar que con esta asignatura he conseguido confiar en mí y ver que echándole ganas, se puede con todo.

Por último me gustaría comentar la gratificante experiencia que he vivido con el uso del Framework de Spring (Spring MVC, Spring Boot, Spring Data y Spring Security). Me ha sorprendido gratamente como Spring puede ayudar a realizar un desarrollo que en un principio puede ser complejo para alguien que no conoce el entorno y hacerlo de una manera tan fácil.

6. Bibliografía

- 1.- Materiales de la UOC (otras asignaturas).
- 2.- Portal Web de la UOC para consultar otros proyectos.
- 3.- Autor: Craig Walls. Título: Spring. Editorial: Anaya. (4ª edición 2015)
- 4.- Autor: Peter Mularien. Título: Spring Security 3. (edición 2010)
- 5.- Consultas páginas web:

Site	Consultado
https://stackoverflow.com/questions/6827752/whats-the-difference-between-component-repository-service-annotations-in	Oct/18
https://www.javacodegeeks.com/2017/11/difference-component-service-controller-repository-spring.html	Oct/2018
https://springbootdev.com/2017/07/31/spring-framework-component-service-repository-and-controller/	Nov/2018
https://www.oracle.com/technetwork/java/javaee/overview/index.html	Dic/2018
http://spring.io/projects	Dic/2018

7. Anexos

7.1. Manual de instalación:

Esta aplicación la he desarrollado en un equipo con sistema operativo Windows 7 de 64 bits. El lenguaje de programación ha sido Java y para su desarrollo he escogido el JDK 1.8. El entorno de trabajo ha sido IntelliJ IDEA 2018 1.4 y como Sistema de Gestión de Base de Datos he utilizado PostgreSQL 9.6.

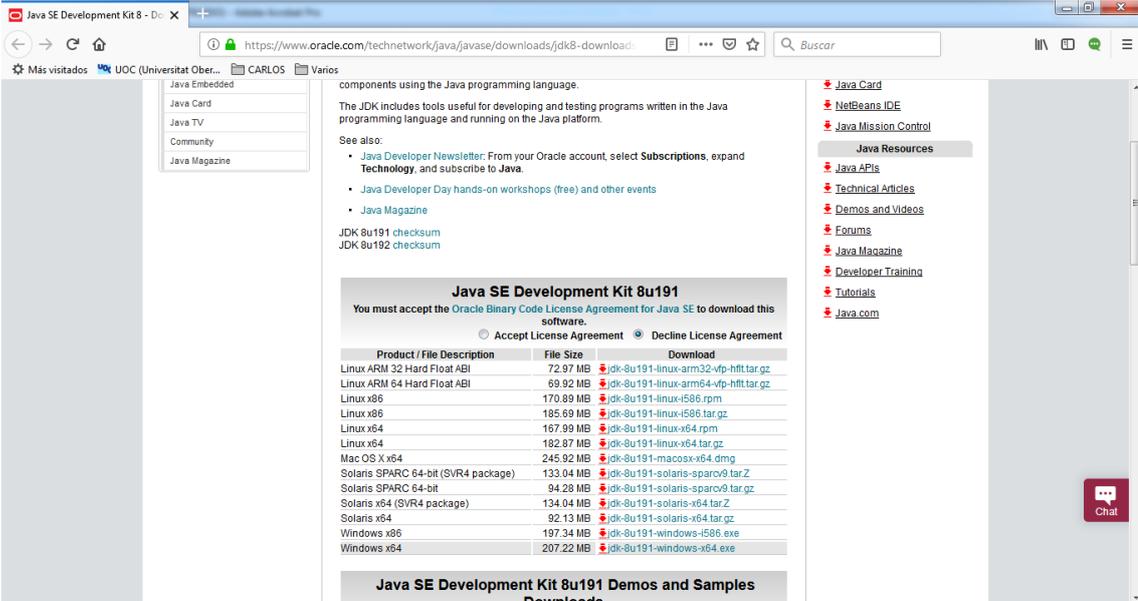
Para poder desplegar esta aplicación hace falta tener instalado los componentes anteriores. Por eso voy a indicar paso a paso como poder instalar cada uno de ellos:

Instalación y configuración de Java:

Cómo el desarrollo lo hemos realizado en Java, para poder ejecutar y compilar nuestra aplicación vamos a necesitar descargar el Java Development Kit (JDK). Como hemos utilizado la versión 1.8, podemos encontrarla en la siguiente url:

<https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html>

Para nuestro entorno que es de 64 bits, procederíamos a descargar la versión de Windows x64



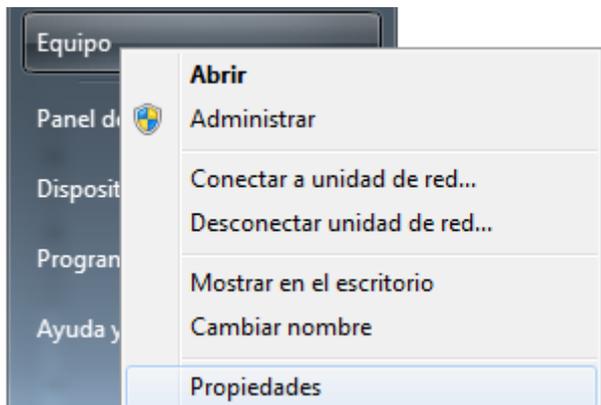
The screenshot shows the Oracle Java SE Development Kit 8u191 download page. The main content area features a table of download links for various operating systems and architectures. The 'Windows x64' row is highlighted, showing a file size of 207.22 MB and a download link 'jdk-8u191-windows-x64.exe'. The page also includes a license agreement section and a sidebar with navigation links.

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	72.97 MB	jdk-8u191-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	69.92 MB	jdk-8u191-linux-arm64-vfp-hflt.tar.gz
Linux x86	170.89 MB	jdk-8u191-linux-i586.rpm
Linux x86	185.69 MB	jdk-8u191-linux-i586.tar.gz
Linux x64	167.99 MB	jdk-8u191-linux-x64.rpm
Linux x64	182.87 MB	jdk-8u191-linux-x64.tar.gz
Mac OS X x64	245.92 MB	jdk-8u191-macosx-x64.dmg
Solaris SPARC 64-bit (SVR4 package)	133.04 MB	jdk-8u191-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	94.28 MB	jdk-8u191-solaris-sparcv9.tar.gz
Solaris x64 (SVR4 package)	134.04 MB	jdk-8u191-solaris-x64.tar.Z
Solaris x64	92.13 MB	jdk-8u191-solaris-x64.tar.gz
Windows x86	197.34 MB	jdk-8u191-windows-i586.exe
Windows x64	207.22 MB	jdk-8u191-windows-x64.exe

Una vez descargado, tendremos que dar doble clic al fichero para lanzar su ejecutable, seguiremos los pasos de instalación y reiniciaremos al terminar en caso de solicitarlo.

Posteriormente, tenemos que configurar las variables de entorno:

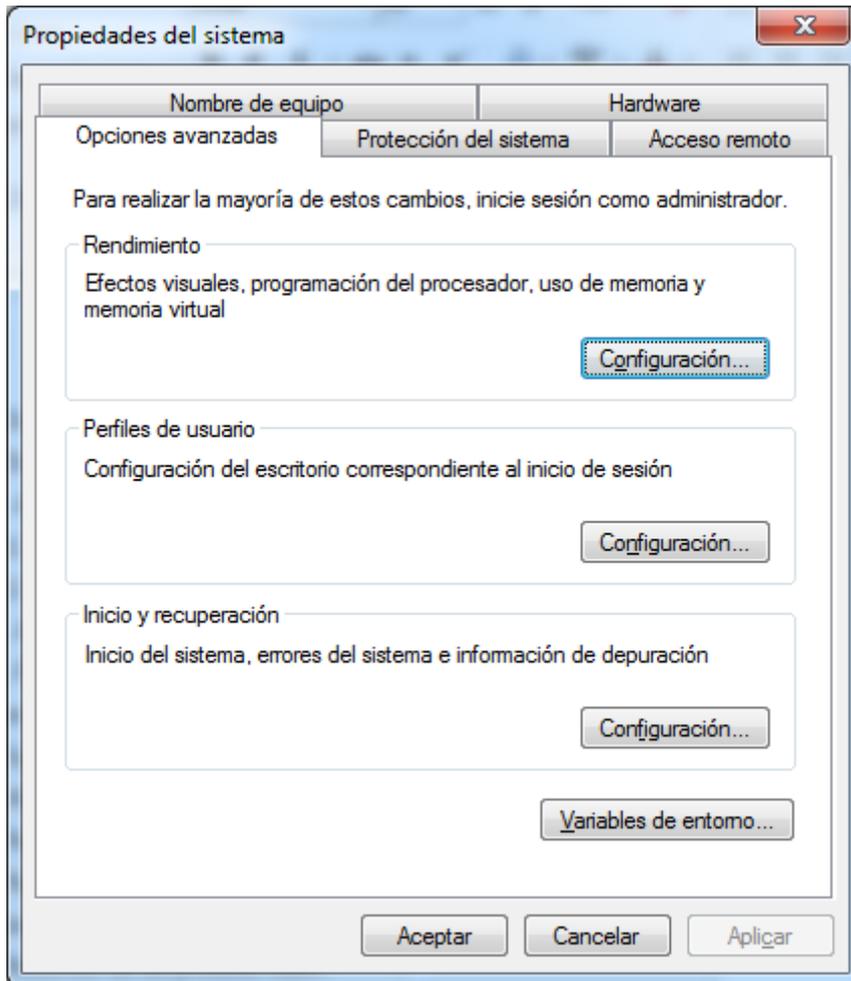
Tenemos que pulsar en el botón de inicio, y en Equipo daremos al botón derecho. Nos aparece el siguiente menú y pulsaremos en propiedades



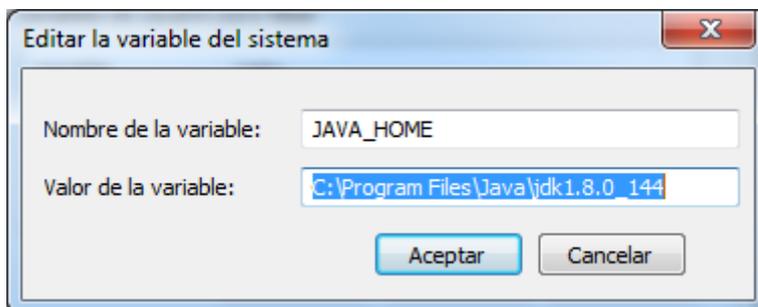
Seleccionaremos Configuración avanzada del sistema, según se muestra en la siguiente imagen:



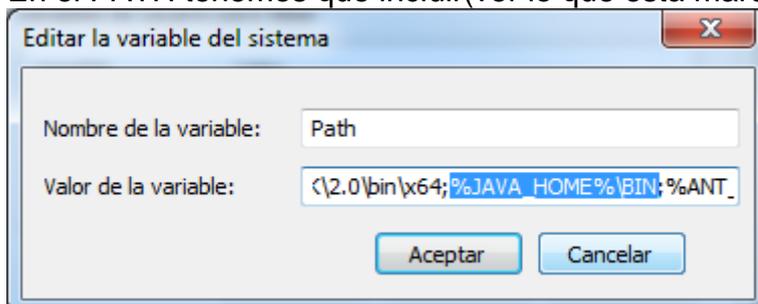
Nos aparecerá la siguiente ventana:

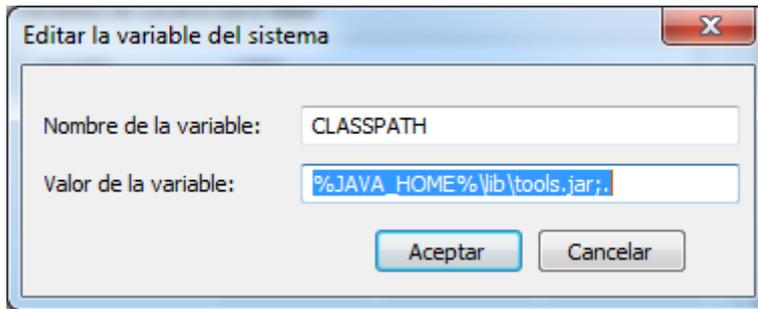


Pulsaremos en Variables de entorno y posteriormente tenemos que definir las variables JAVA_HOME y configura el PATH y CLASSPATH



En el PATH tenemos que incluir(ver lo que está marcado en azul):





Instalación y configuración de Java:

Posteriormente tenemos que instalar el SGBD. En nuestro caso es PostgreSQL en la versión 9.6 la versión de 64 bits. Para descargar la aplicación podemos hacerlo en la siguiente url:

<https://www.postgresql.org/download/>

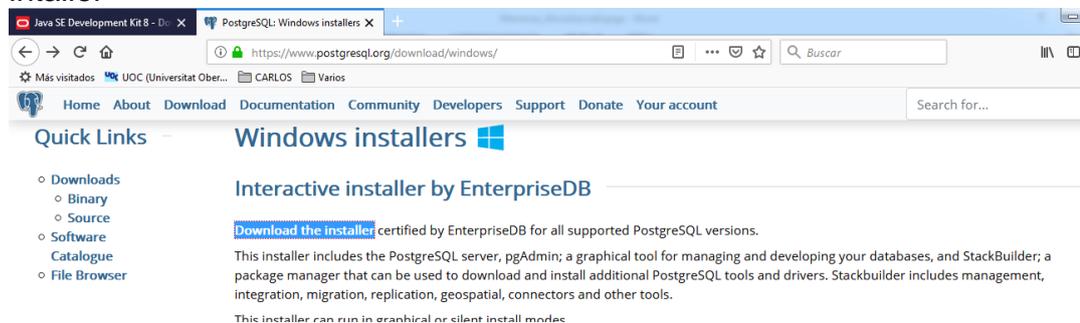
Cómo está instalado en un equipo Windows, seleccionaremos el S.O. para descargar el instalable:

Binary packages

Pre-built binary packages are available for a number of different operating systems:

- BSD
 - FreeBSD
 - OpenBSD
- Linux
 - Red Hat family Linux (including CentOS/Fedora/Scientific/Oracle variants)
 - Debian GNU/Linux and derivatives
 - Ubuntu Linux and derivatives
 - SuSE and OpenSuSE
 - Other Linux
- macOS
- Solaris
- Windows

Se abrirá una nueva página donde tendremos que pulsar en Download de intaller



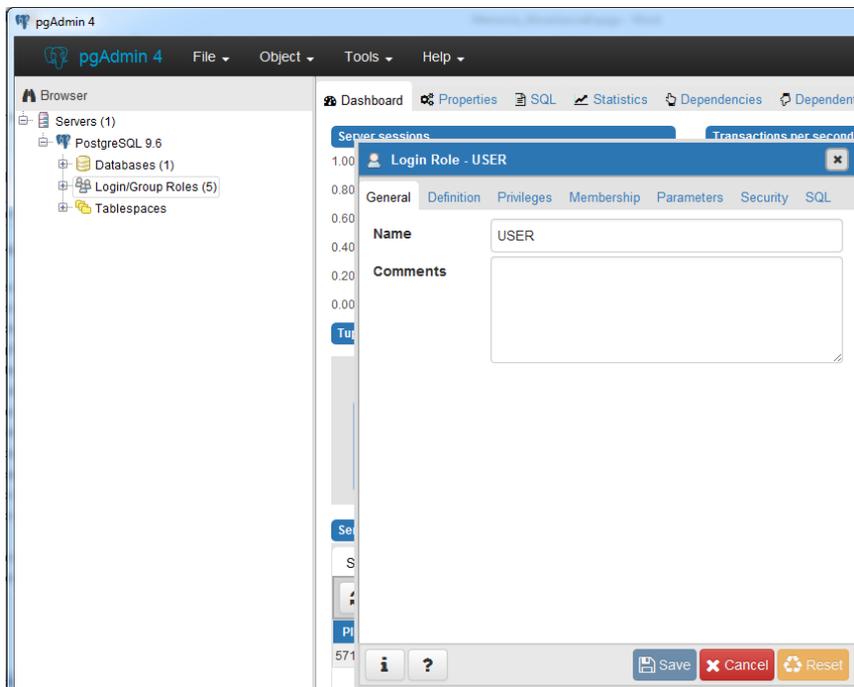
Y por último seleccionaremos el sistema operativo y la versión correcta de postgresQL

PostgreSQL Download

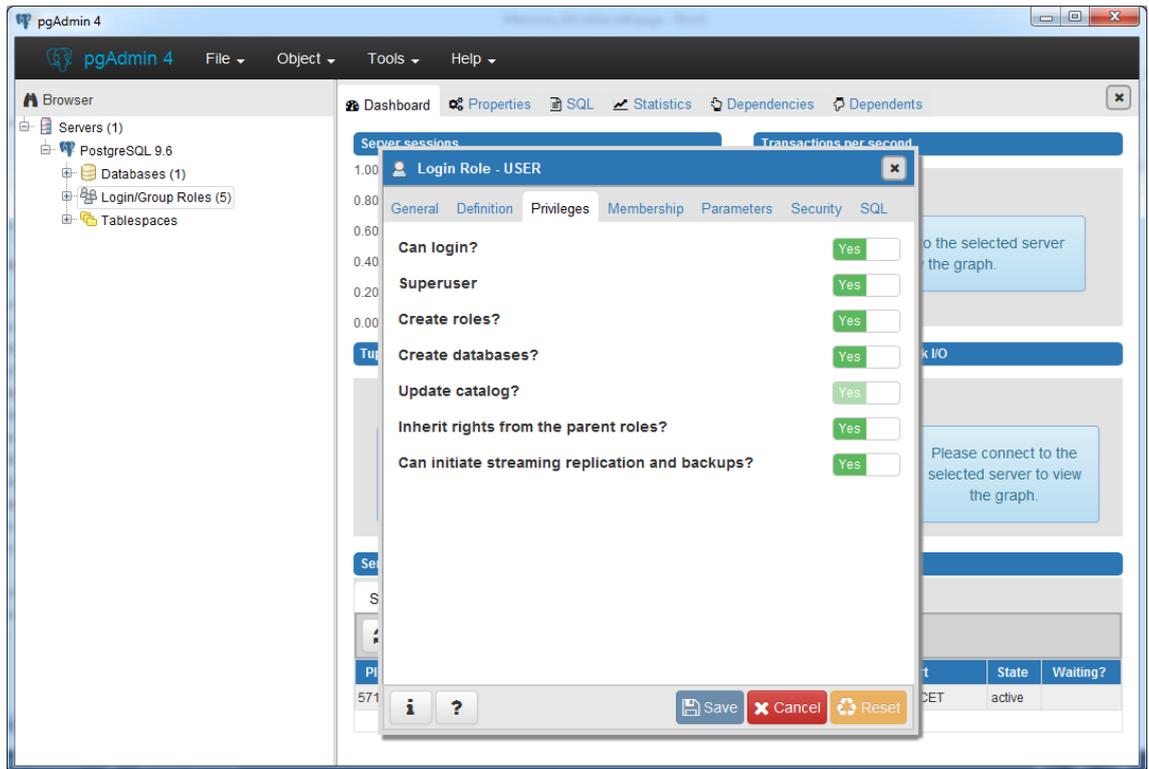
PostgreSQL Version	Linux x86-64	Linux x86-32	Mac OS X	Windows x86-64	Windows x86-32
11.1	N/A	N/A	Download	Download	N/A
10.6	Download				
9.6.11	Download				
9.5.15	Download				

También pulsaremos doble clic en el instalable y seguiremos los pasos para su instalación standard.

Una vez instalado, abriremos la consola de postgres (pgAdmin) y tendremos que crear un rol de acceso que tendrá como nombre de usuario : USER y de password : PASSWORD.



Tenemos que darle todos los privilegios:



Y lo guardamos.

Posteriormente tenemos que crear la base de datos y el esquema donde guardaremos los datos. Para ello podemos crearla usando pgAdmin y en caso de que tengamos creada la base de datos, podríamos ejecutar el siguiente script:

```
CREATE USER cinemama PASSWORD 'password';
```

```
ALTER USER cinemama WITH LOGIN;
```

```
CREATE DATABASE postgres WITH OWNER cinemama;
```

```
CREATE SCHEMA cinemama AUTHORIZATION cinemama
```

Para tener un banco de datos con el que realizar pruebas, también podríamos lanzar el siguiente script que creará las tablas de nuestro sistema e introducirá datos en ellas para realizar pruebas:

```
DROP TABLE IF EXISTS CINEMAMA.EVAL;
```

```
DROP TABLE IF EXISTS CINEMAMA.PELICULA;
```

```
DROP TABLE IF EXISTS CINEMAMA.CINE;
```

```
DROP TABLE IF EXISTS CINEMAMA.USUARIO_ACCESO;
```

```
CREATE TABLE CINEMAMA.EVAL(id serial PRIMARY KEY,
COMENTARIO VARCHAR(100), VALOR integer, ID_USUARIO bigint,
ID_PELICULA bigint, VERSION integer default 0);
```

```

ALTER TABLE CINEMAMA.EVAL OWNER to cinemama;
INSERT INTO CINEMAMA.EVAL(COMENTARIO, VALOR,
ID_USUARIO,ID_PELICULA,VERSION) VALUES('Comentario ADMIN
PELI 1', 2,1,1,0);
INSERT INTO CINEMAMA.EVAL(COMENTARIO, VALOR,
ID_USUARIO,ID_PELICULA,VERSION) VALUES('Comentario
USUARIO1 PELI 1', 8,2,1,0);
INSERT INTO CINEMAMA.EVAL(COMENTARIO, VALOR,
ID_USUARIO,ID_PELICULA,VERSION) VALUES('Comentario ADMIN
PELI 2', 4,1,2,0);
INSERT INTO CINEMAMA.EVAL(COMENTARIO, VALOR,
ID_USUARIO,ID_PELICULA,VERSION) VALUES('Comentario ADMIN
PELI 3', 8,1,3,0);
INSERT INTO CINEMAMA.EVAL(COMENTARIO, VALOR,
ID_USUARIO,ID_PELICULA,VERSION) VALUES('Comentario
USUARIO1 PELI 3', 4,2,3,0);
INSERT INTO CINEMAMA.EVAL(COMENTARIO, VALOR,
ID_USUARIO,ID_PELICULA,VERSION) VALUES('Comentario
USUARIO2 PELI 3', 7,3,3,0);

```

```

CREATE TABLE CINEMAMA.PELICULA(id serial PRIMARY KEY,
CLASIFICACION INTEGER,EDADRECOMENDADA INTEGER,
PUBLICACION VARCHAR(4), SINOPSIS VARCHAR(255),TITULO
VARCHAR(100),VALORACION REAL, VERSION integer default 0);
ALTER TABLE CINEMAMA.PELICULA OWNER TO cinemama;
INSERT INTO CINEMAMA.PELICULA(TITULO,
CLASIFICACION,EDADRECOMENDADA, PUBLICACION, SINOPSIS,
VALORACION,VERSION) VALUES('GHOST', 4,0, '1996', 'Un hombre es
asesinado y su fantasma ayuda a su novia', 5.0,0);
INSERT INTO CINEMAMA.PELICULA(TITULO,
CLASIFICACION,EDADRECOMENDADA, PUBLICACION, SINOPSIS,
VALORACION,VERSION) VALUES('TOY STORY', 3,1, '2001', 'Historia
de unos juguetes', 4.0,0);
INSERT INTO CINEMAMA.PELICULA(TITULO,
CLASIFICACION,EDADRECOMENDADA, PUBLICACION, SINOPSIS,
VALORACION,VERSION) VALUES('LA JUNGLA DE CRISTAL', 0,4,
'2000', 'Un agente tiene que salvar el mundo', 6.33333,0);

```

```

CREATE TABLE CINEMAMA.CINE(id serial PRIMARY KEY,
AFOROSALA VARCHAR(255),DIRECCION
VARCHAR(255),HORASESION VARCHAR(255),NOMBRE
VARCHAR(255),PROVINCIA VARCHAR(255),TELEFONO
VARCHAR(255),ID_PELICULA bigint,VERSION INTEGER default 0);
ALTER TABLE CINEMAMA.CINE OWNER TO cinemama;
INSERT INTO
CINEMAMA.CINE(AFOROSALA,DIRECCION,HORASESION,NOMBRE,
PROVINCIA,TELEFONO,ID_PELICULA,VERSION)
VALUES('45','CALLE ALCALA','11:00','CINE
ALACALA','MADRID','915555555',3,0);

```

```

INSERT INTO
CINEMAMA.CINE(AFOROSALA,DIRECCION,HORASESION,NOMBRE,
PROVINCIA,TELEFONO,ID_PELICULA,VERSION) VALUES('70','AVDA.
ILUSTRACION','10:00','CINE VAGUADA','MADRID','916665555',3,0);
INSERT INTO
CINEMAMA.CINE(AFOROSALA,DIRECCION,HORASESION,NOMBRE,
PROVINCIA,TELEFONO,ID_PELICULA,VERSION) VALUES('55','GRAN
VIA','11:00','CINE GRAN VIA','TOLEDO','925009877',2,0);
INSERT INTO
CINEMAMA.CINE(AFOROSALA,DIRECCION,HORASESION,NOMBRE,
PROVINCIA,TELEFONO,ID_PELICULA,VERSION)
VALUES('55','CALLE CUENCA 24','11:30','CINE
CUENCA','CUENCA','999222222',1,0);

```

```

CREATE TABLE CINEMAMA.USUARIO_ACCESO(id serial PRIMARY
KEY, USUARIO VARCHAR(10) NOT NULL, PASSWORD
VARCHAR(10), ROL INTEGER, VERSION integer default 0);
ALTER TABLE CINEMAMA.USUARIO_ACCESO OWNER TO
cinemama;
INSERT INTO CINEMAMA.USUARIO_ACCESO(USUARIO,
PASSWORD, ROL, VERSION) VALUES('ADMIN','ADMIN',0,0);
INSERT INTO CINEMAMA.USUARIO_ACCESO(USUARIO,
PASSWORD, ROL, VERSION) VALUES('USER1','USER1',1,0);
INSERT INTO CINEMAMA.USUARIO_ACCESO(USUARIO,
PASSWORD, ROL, VERSION) VALUES('USER2','USER2',1,0);

```

En nuestro caso, al usar Maven y Spring Boot no nos hace falta configura un servidor de aplicaciones ni los conectores con BBDD, no obstante si se quisiera modificar algún parámetro de los que he mencionado en este manual, se podría cambiar y en nuestro proyecto deberíamos actualizar el fichero application.properties. En el apartado del fichero en el que hago mención a la conexión con la base de datos es donde he definido el usuario y password de acceso a la misma de la siguiente manera:

```

#####
# PROPIEDADES DE LA BASE DE DATOS #
#####
spring.datasource.platform=postgres
spring.datasource.url=jdbc:postgresql://localhost:5432/postgres
spring.datasource.username=cinemama
spring.datasource.password=password

```

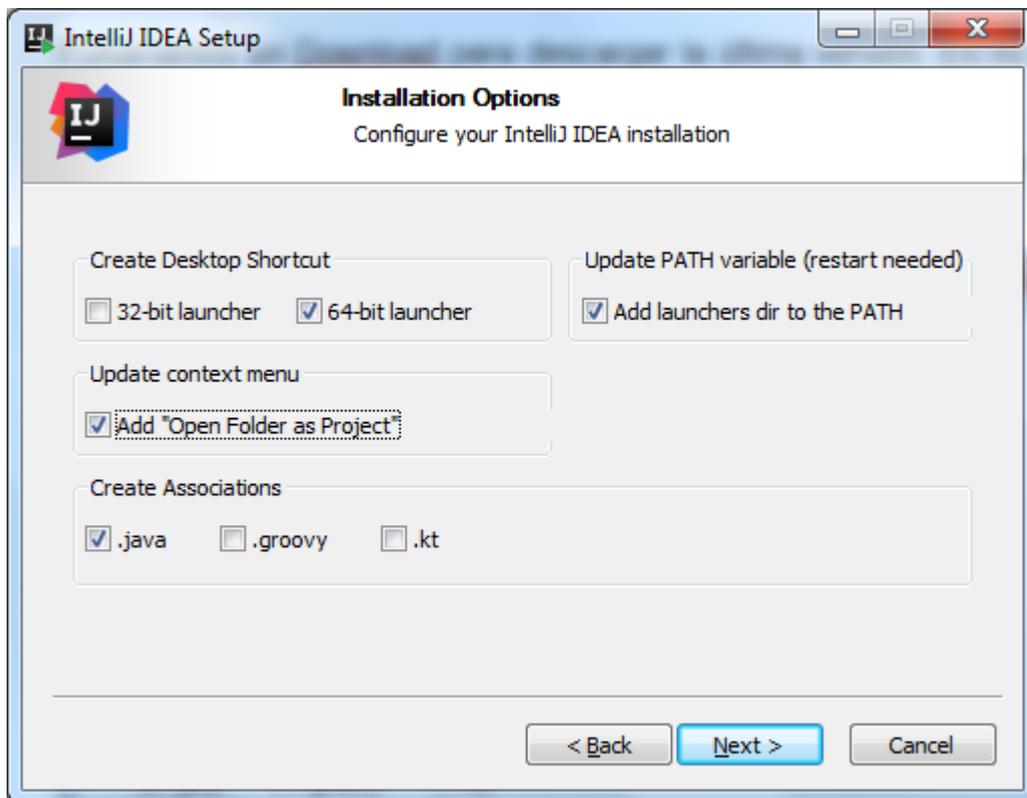
Instalación y configuración del IDE:

Como entorno de trabajo he utilizado IntelliJIDEA y podemos descargarlo en :

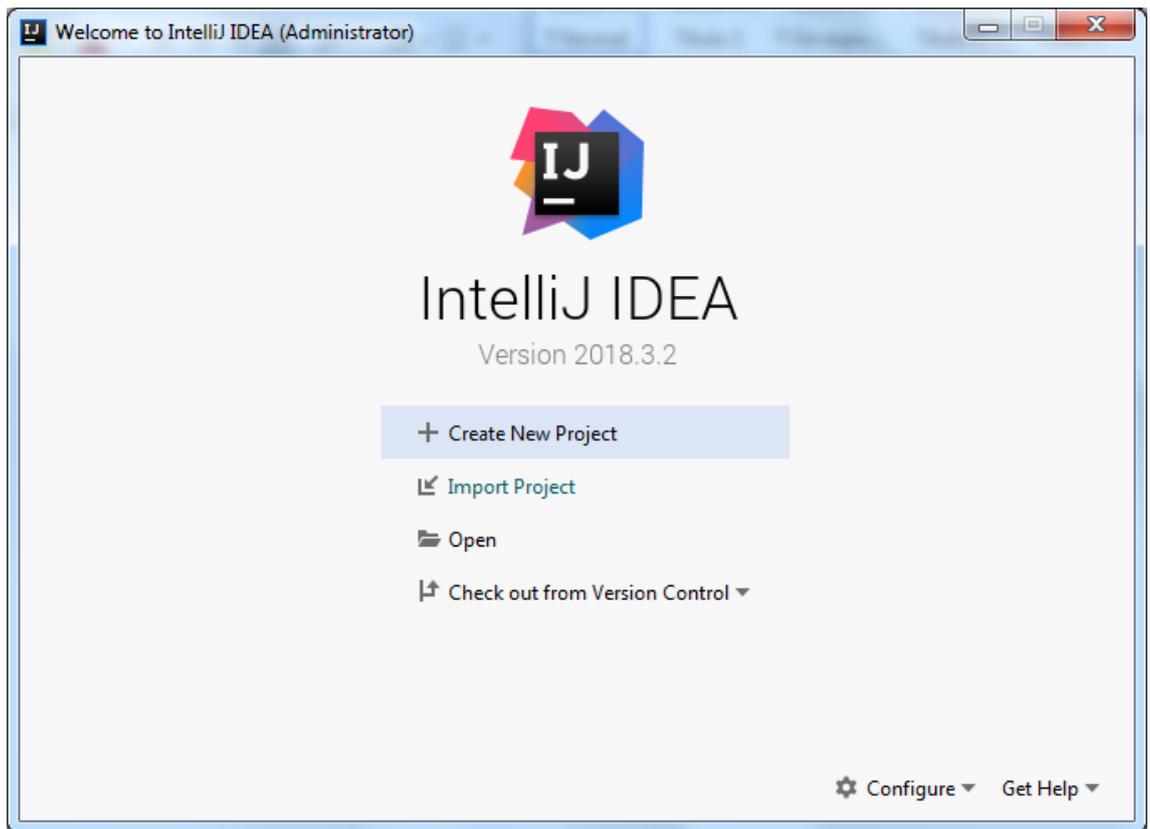
<https://www.jetbrains.com/idea/?fromMenu>

Pulsaríamos en Download para descargar la última versión. En mi caso tengo instalada la versión 1.4 de 64bits. (Versión ultimate)

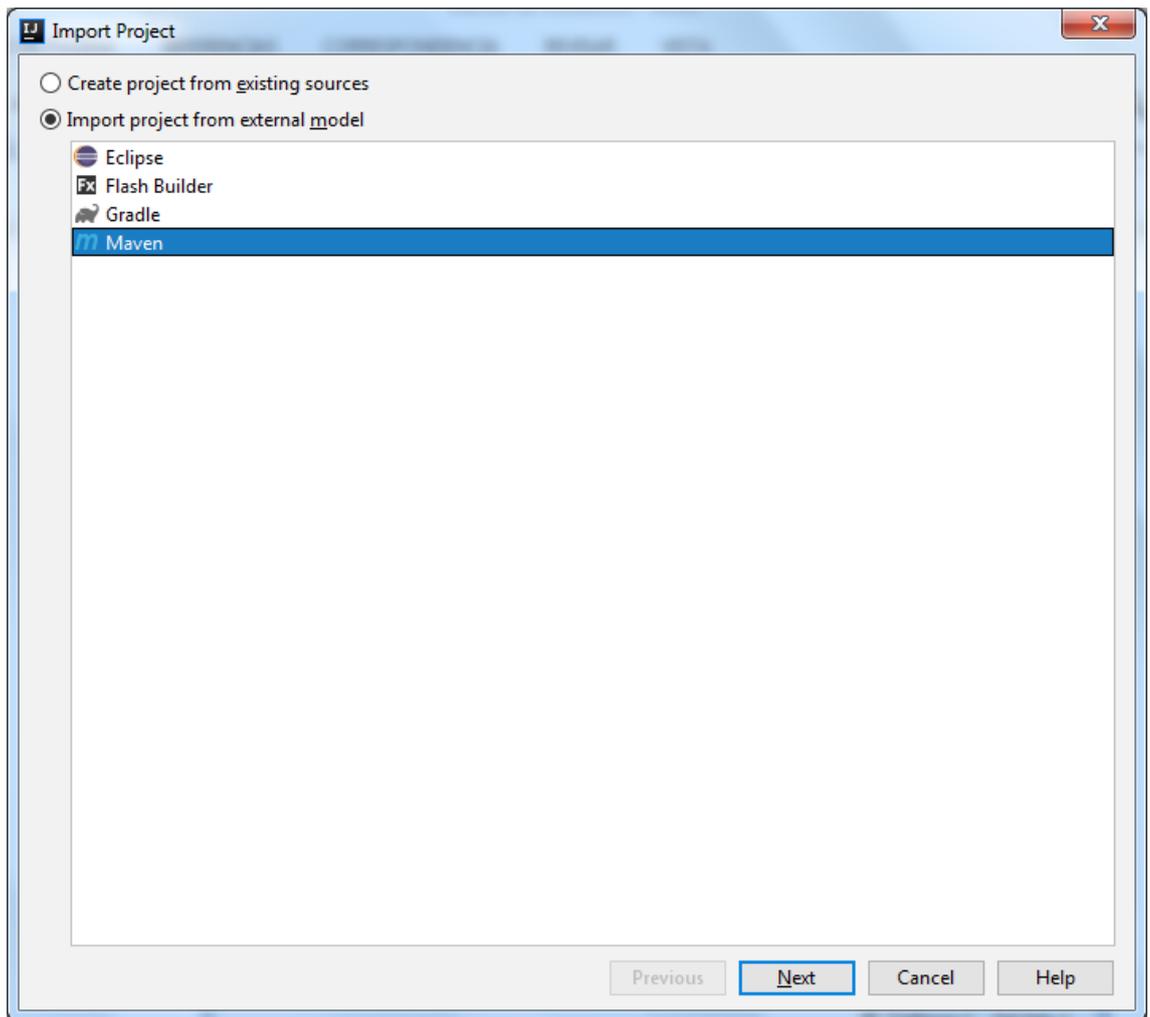
Al igual que en los casos anteriores, seguiremos en instalable y nos instalará el producto indicado.



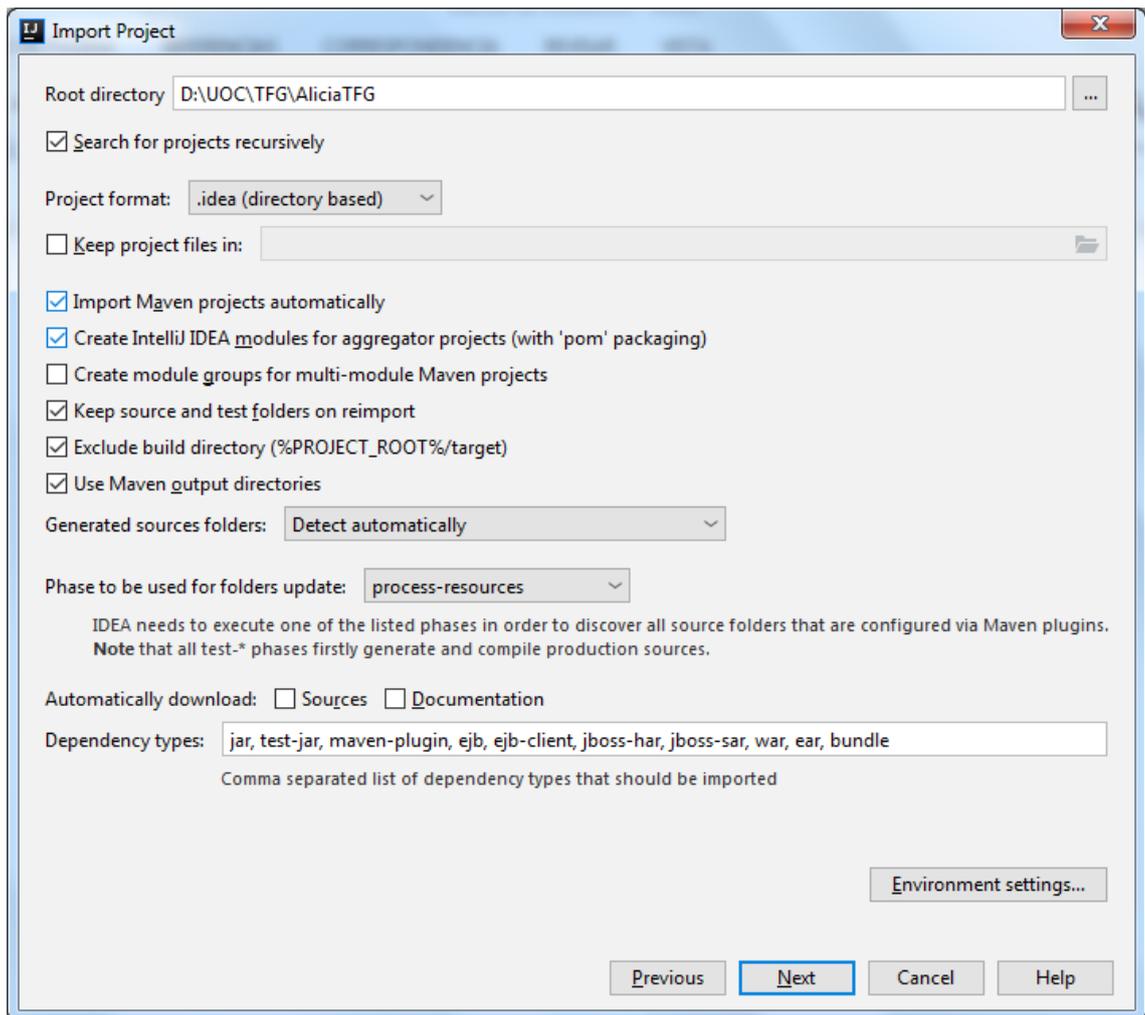
En la siguiente ventana, seleccionar import Project.



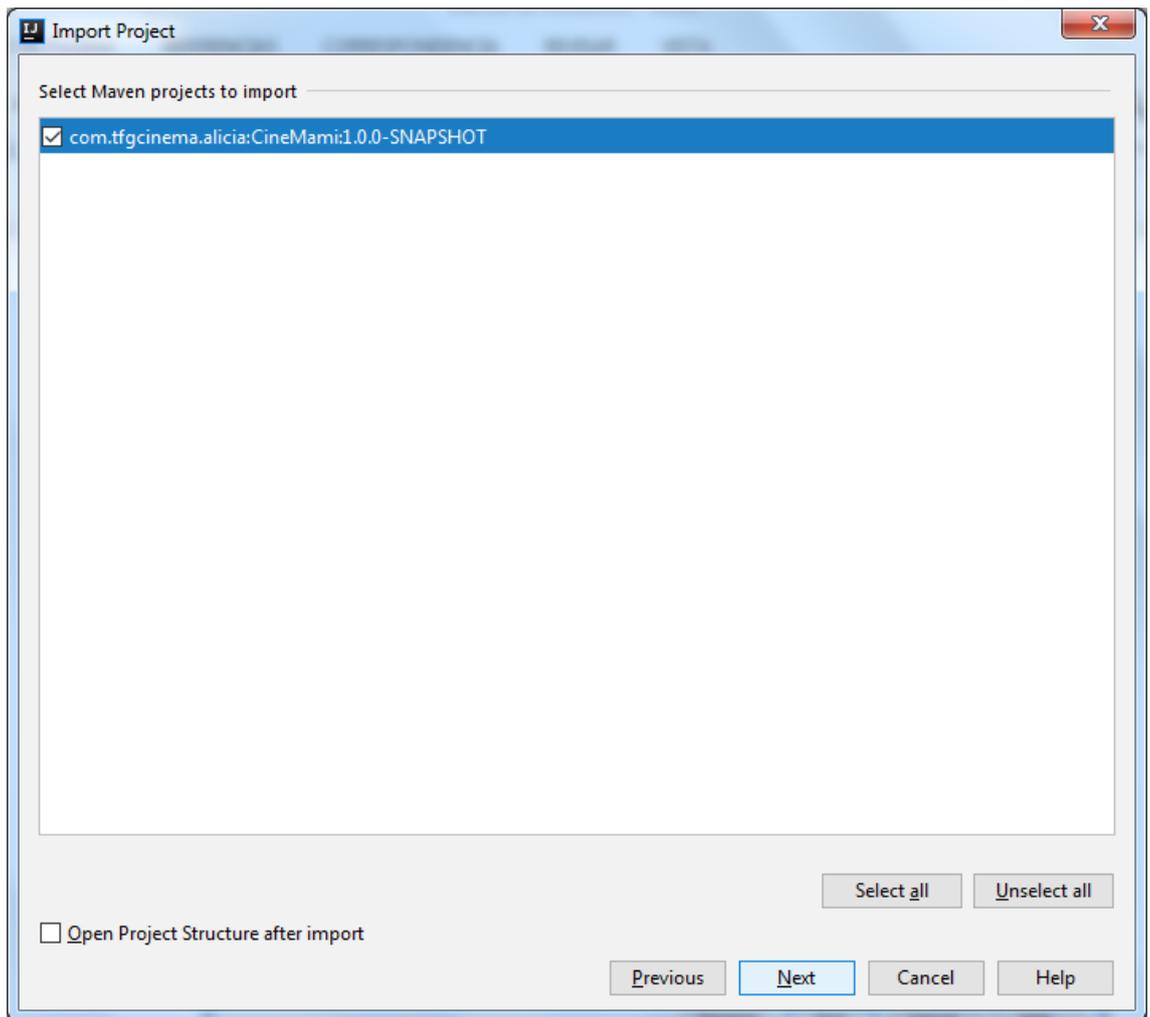
Importamos el proyecto desde un modelo externo, y seleccionamos maven.



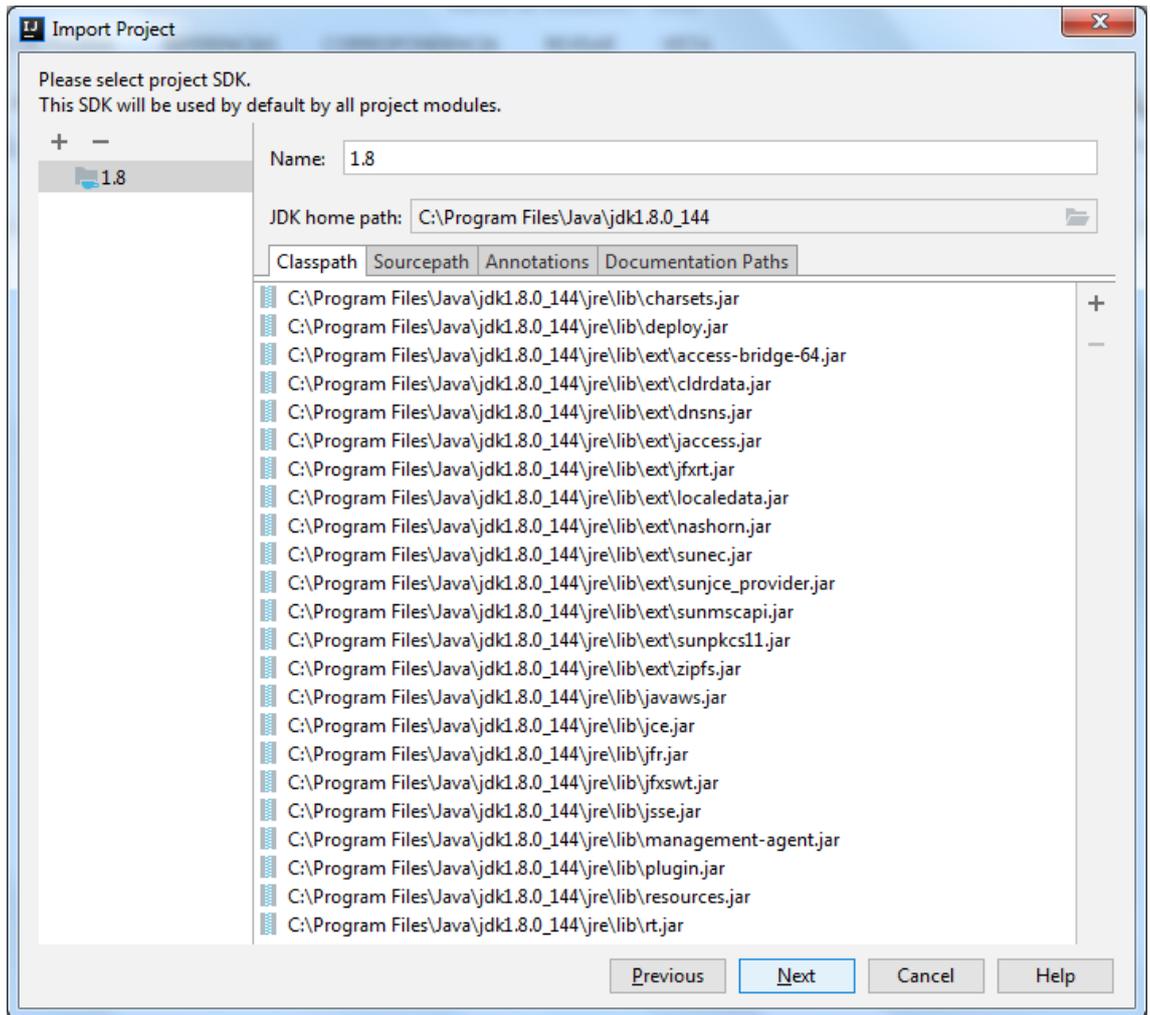
Marcamos las opciones tal y como muestro en la siguiente ventana (hay que seleccionar la ruta del proyecto y los checks indicados):



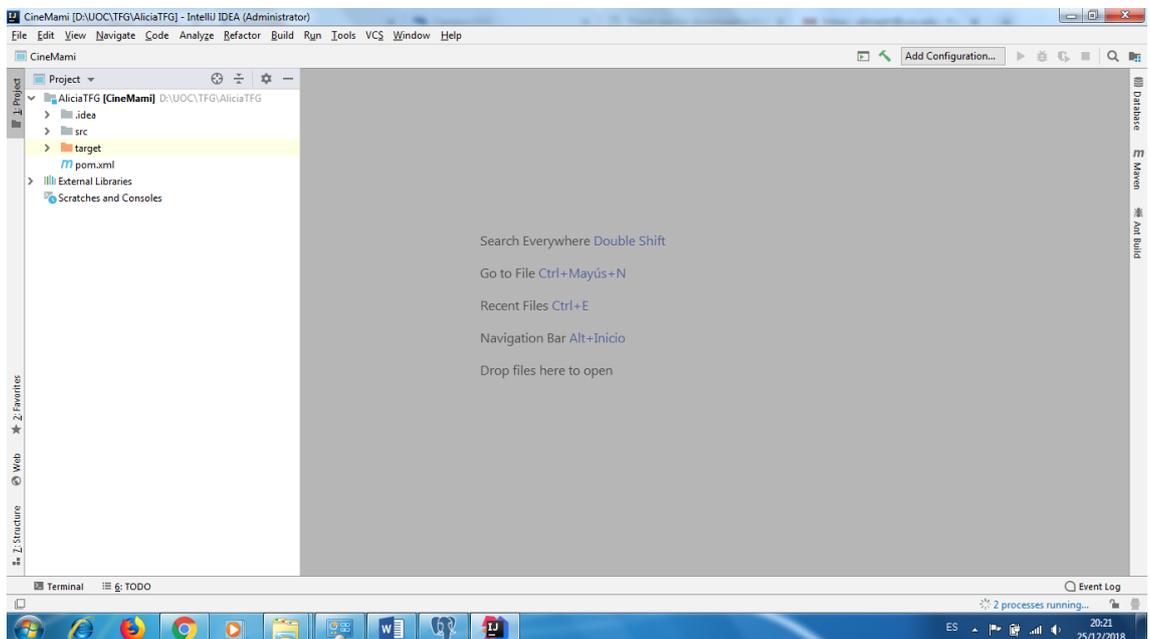
Vendrá por defecto mi proyecto seleccionado:



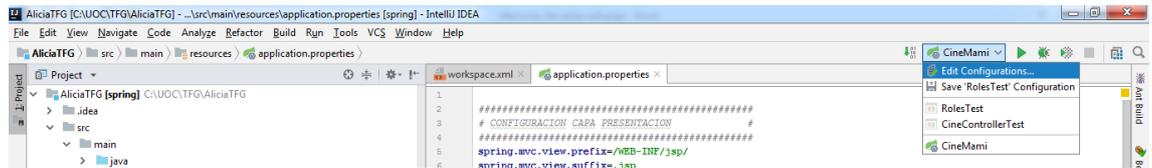
Vemos cómo se cargan las librerías:



Esperar a que importe todo (esquina inferior derecha debe terminar de procesar todo).



En el caso en el que no se cree automáticamente una configuración, habría que crearla para el uso de Spring Boot. Para ello pulsamos en la esquina superior derecha y seleccionamos edit Configurations



Tenemos que decir a nuestro entorno que vamos a usar Spring boot, por los que lo añadimos pulsando en el botón + y le indicamos la clase que va a ser la principal "com.tfgcinema.alicia.CineMami". Guardamos y le damos a ok.

Para ver si está correcto pulsaremos en el botón play de la barra superior derecha:

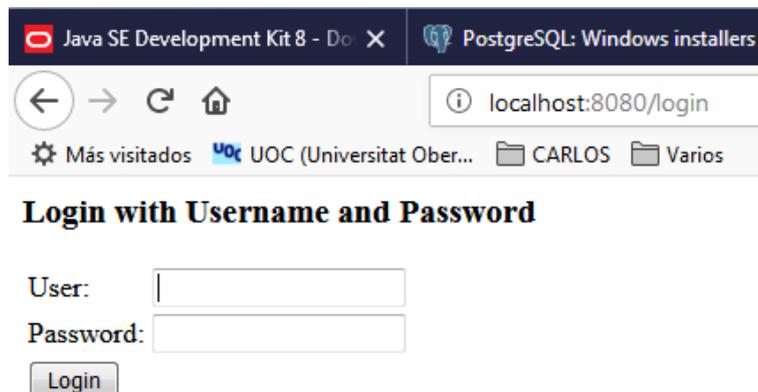


En la consola, pasados unos segundos podremos ver un mensaje indicándonos que se ha cargado correctamente:

```
2018-12-22 23:36:22.456 INFO 6428 --- [ restartedMain]
com.tfgcinema.alicia.CineMami : Started CineMami in 75.262
seconds (JVM running for 118.191)
```

Para ver si funciona tendremos que abrir un navegador y abrir la url:

localhost:8080



Las credenciales de acceso son las siguientes:

ROL Administrador:
User : ADMIN
Password: ADMIN

ROL Usuario:
User: USER1
Password: USER1

ROL Usuario:
User: USER2
Password: USER2