



CineNova

Especificación de Requisitos

Versión: 0102

Fecha: 22/05/2025

0102

Queda prohibido cualquier tipo de explotación y, en particular, la reproducción, distribución, comunicación pública y/o transformación, total o parcial, por cualquier medio, de este documento sin el previo consentimiento expreso del autor

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

HOJA DE CONTROL

Organismo	Consejería		
Proyecto	CineNova		
Entregable	Especificación de Requisitos		
Autor	1º DAW		
Versión/Edición	0102	Fecha Versión	12/05/2025
Aprobado por	Consejería	Fecha Aprobación	22/05/2025
		Nº Total de Páginas	30

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
0100	Versión inicial	Juan Carlos Pizarro Alonso	10/05/2025
0101	Versión avanzada	Álvaro Muñoz Fernández	12/05/2025
0102	Versión final	Juan Carlos Pizarro Alonso y Álvaro Muñoz Fernández	22/05/2025

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
Juan Carlos Pizarro Alonso
Álvaro Muñoz Fernández

ÍNDICE

	<p style="text-align: center;">CineNova Especificación de Requisitos</p>	<p style="text-align: center;">IES.CAMAS</p>
---	--	---

1 INTRODUCCIÓN.....	5
1.1 Alcance.....	5
1.2 Objetivos.....	5
2 INFORMACIÓN DEL DOMINIO DEL PROBLEMA.....	6
2.1 Introducción al Dominio del Problema.....	6
2.2 Glosario de Términos.....	6
3 DESCRIPCIÓN DE LA SITUACIÓN ACTUAL (¿Qué problema solucionaría?).....	7
3.1 Pros y Contras de la Situación Actual.....	7
4 NECESIDADES DE NEGOCIO.....	8
4.1 Objetivos de Negocio.....	8
5 MODELADO DE REQUISITOS.....	9
5.1 Casos de Uso (Análisis).....	9
5.2 Requisitos Funcionales.....	10
5.3 Requisitos No Funcionales.....	10
6 Modelado del Sistema (Diseño).....	11
6.1 Diagrama de clases.....	11
6.2 Diagrama Entidad-Relación.....	11
6.3 Diagrama de Actividad.....	11
6.4 Diagrama de Estados (Opcional).....	11
7 Plan de pruebas y validación.....	12
7.1 Plan de Pruebas Unitarias.....	12
7.2 Trazabilidad de Requisitos.....	12
8 Gestión configuración.....	13
8.1 Estructura del repositorio.....	13
8.2 Gestor del Ciclo de Vida.....	13
9 ANEXOS [OPCIONAL].....	14
9.1 Anexo A: Actas de Reuniones.....	14
9.2 Anexo B: Documentación Relevante.....	14
9.3 Anexo C: Glosario de Acrónimos y Abreviaturas.....	14

	<p align="center">CineNova</p> <p align="center">Especificación de Requisitos</p>	<p align="center">IES.CAMAS</p>
---	---	--

1 INTRODUCCIÓN

El objetivo principal es establecer los requisitos para el desarrollo de un sistema de gestión de un cine, que permita de forma digital la administración de los empleados por parte de los jefes, las películas, salas y sesiones por parte de los empleados y las entradas por parte de los clientes.

1.1 Alcance

El sistema cubrirá las siguientes funcionalidades:

- Los jefes del cine podrán:
 - Añadir, modificar y eliminar empleados del sistema.
- Los empleados y jefes del cine podrán:
 - Añadir, modificar y eliminar películas de la cartelera.
 - Añadir y eliminar sesiones (asignación de una película a una sala en una hora específica).
- Los clientes del cine podrán:
 - Registrarse y autenticarse en el sistema
 - Visualizar las sesiones disponibles.
 - Comprar entradas asignadas a una sesión y asiento concreto.
 - Cancelar sus entradas si es necesario.
 - Descargar un recibo de sus entradas si lo desean.

1.2 Objetivos

Los objetivos principales del sistema son:

- Facilitar la gestión de las películas y sesiones de forma digital para los empleados y la gestión de los empleados para los jefes.
- Permitir a los clientes poder comprar, devolver y/o descargar un recibo de sus entradas a las diferentes sesiones de forma digital.
- Poder aumentar la eficacia del cine, disminuyendo errores manuales y centralizando toda la gestión.
- Tener un registro histórico de forma digital para futuras consultas.

 CineNova	<p style="text-align: center;">CineNova Especificación de Requisitos</p>	<p style="text-align: center;">IES.CAMAS</p>
--	--	---

2 INFORMACIÓN DEL DOMINIO DEL PROBLEMA

El presente sistema se aplicará al entorno de gestión de cines, orientado tanto a la administración interna como a la interacción con los clientes. Permitirá modernizar los procesos actualmente utilizados en el cine para llevar un control organizado y eficaz de las actividades principales, siendo más útil tanto para empleados/jefes como para clientes.

2.1 Introducción al Dominio del Problema

En la actualidad, la gestión del cine se realiza mediante hojas de cálculo, lo que resulta poco eficiente, especialmente en la planificación de salas, sesiones y películas. Además, la compra de entradas es exclusivamente presencial, lo que limita la comodidad para los clientes y genera colas, errores y dificultad.

El sistema propuesto digitaliza todas estas operaciones, ofreciendo una solución integrada que mejora la accesibilidad, el control de datos y la experiencia del cliente.

2.2 Glosario de Términos

Sala = lugar donde se proyectan películas

Película = proyección que cuenta una historia de forma gráfica

Sesión = conjunto de una película y una sala en una fecha y hora concreta

Entrada = permiso de acceso a una sesión determinada y un asiento asignado

Jefe = persona con permisos especiales para gestionar el cine y los empleados

Empleado = persona con permisos especiales para gestionar el cine

Cliente = persona que accede al sistema para comprar entradas y gestionarl

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

3 DESCRIPCIÓN DE LA SITUACIÓN ACTUAL (¿Qué problema solucionaría?)

Actualmente, el cine realiza la gestión de su actividad diaria mediante el uso de hojas de cálculo (por ejemplo, Excel), donde se almacena información sobre las películas disponibles, las salas activas y las sesiones programadas.

Además, las entradas para los clientes se emiten exclusivamente en formato físico en papel, lo que implica que todas las compras y modificaciones deben realizarse de manera presencial en taquilla, generando incomodidades para ambas partes.

3.1 Pros y Contras de la Situación Actual

Pros	Contras
Coste bajo	Gasto de papel innecesario
El cliente puede llevar la entrada física	El cliente debe acudir al cine para devolver o comprar entradas
Sistema conocido y fácil de usar	No hay control en tiempo real
Entrada tangible que no necesita ningún dispositivo	No se permite ni gestión ni compra remota

A pesar de algunas ventajas relacionadas con la simplicidad y el bajo coste, el sistema actual no satisface las necesidades modernas de eficiencia, sostenibilidad y comodidad tanto para empleados como para clientes.

	<p align="center">CineNova</p> <p align="center">Especificación de Requisitos</p>	<p align="center">IES.CAMAS</p>
---	---	--

4 NECESIDADES DE NEGOCIO

El negocio requiere una transformación digital para mejorar la eficiencia operativa y adaptarse a las expectativas actuales de los usuarios, quienes demandan servicios accesibles desde dispositivos digitales, como la compra online de entradas o la consulta de la cartelera desde casa.

Con un sistema centralizado y automatizado, el cine podrá optimizar sus recursos humanos, reducir errores administrativos, ofrecer un mejor servicio al cliente y generar registros útiles para análisis y toma de decisiones.

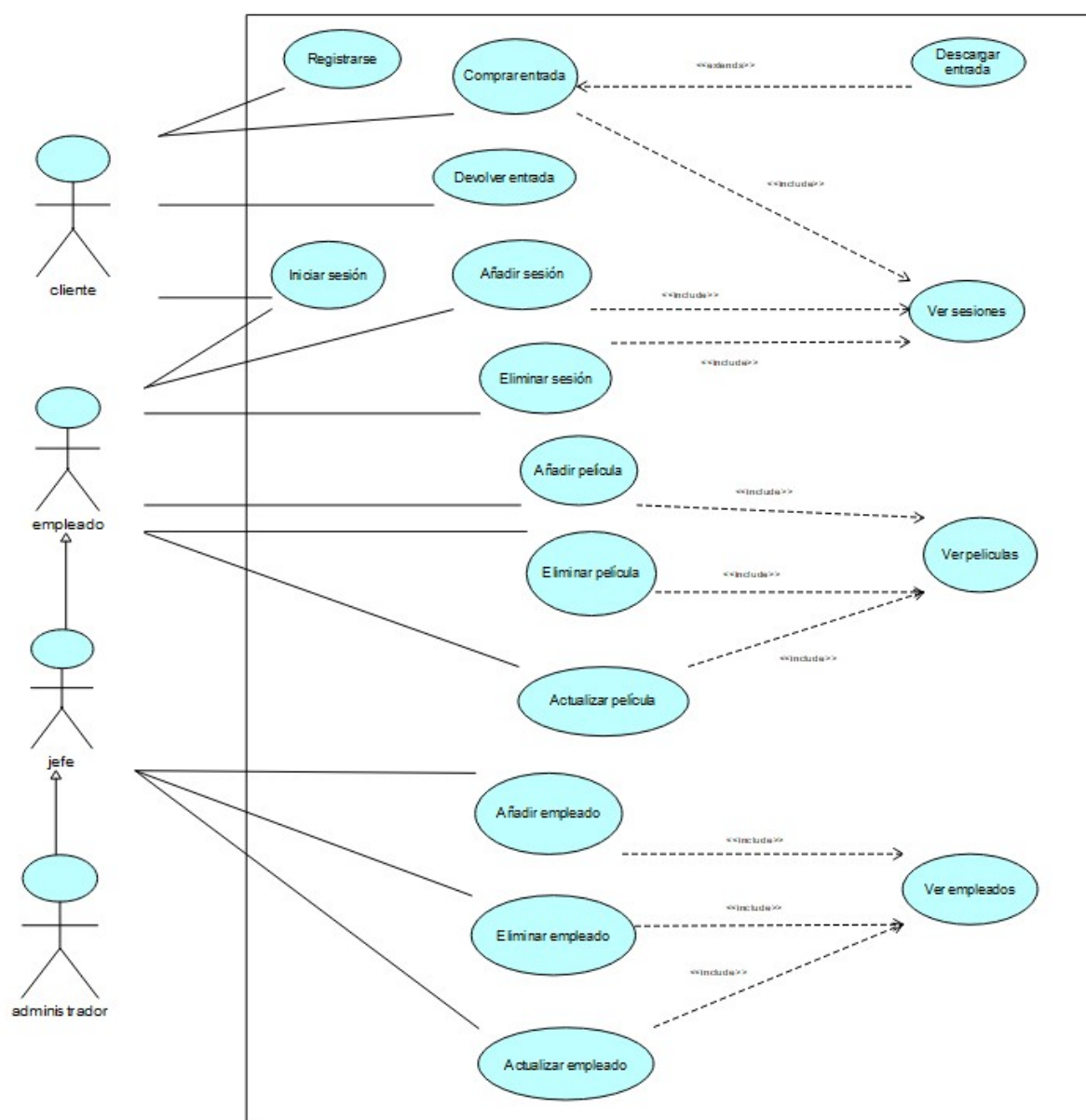
4.1 Objetivos de Negocio

- Mejorar la experiencia y comodidad del cliente a la hora de adquirir entradas.
- Digitalizar la gestión de sesiones y películas, reemplazando el uso de hojas de cálculo.
- Ofrecer un sistema de compra de entradas en línea, disponible en cualquier momento para los clientes.
- Reducir el trabajo manual y los errores administrativos, especialmente en la programación de sesiones.
- Aumentar la satisfacción del cliente al proporcionar una experiencia más rápida, cómoda y moderna.
- Disminuir el uso de papel, contribuyendo a prácticas más sostenibles y eficientes.

 CineNova	<p align="center">CineNova Especificación de Requisitos</p>	<p align="center">IES.CAMAS</p>
--	---	--

5 MODELADO DE REQUISITOS

5.1 Casos de Uso (Análisis)



 CineNova	CineNova Especificación de Requisitos	IES.CAMAS
--	--	------------------

1. Nombre del caso de uso del sistema		Actualizar empleado
2. ID del Caso de uso		CU-13
3. Descripción del caso de uso		
El jefe del sistema accede a la información de un empleado existente y modifica uno o dos de sus datos (Contraseña o sueldo). Los cambios se guardan y se reflejan en la base de datos.		
4. Actor(es)		
Jefe		
5. Precondiciones		
<ul style="list-style-type: none"> - El jefe ha iniciado sesión correctamente en el sistema. - El empleado que se desea modificar ya existe en la base de datos. 		
6. Postcondiciones		
<ul style="list-style-type: none"> - La información del empleado ha sido actualizada correctamente. - El sistema refleja los nuevos datos al consultar la información del empleado. 		
7. Flujo principal		
<i>[Pasos que describen la realización del caso de uso. Empieza con la primera acción del actor y el sistema emitirá una respuesta]</i>		
Nº	Acción del actor	Respuesta del sistema
1	El jefe selecciona el empleado a modificar.	El sistema muestra la información actual del empleado.
2	El jefe edita los datos del empleado.	El sistema valida los datos introducidos.
3	El jefe confirma la actualización.	El sistema guarda los cambios

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

7b. Flujo alternativo

[Pasos que describen la realización del caso de uso alternativo]

Nº	Acción del actor	Respuesta del sistema
1	El jefe introduce datos inválidos.	El sistema muestra un mensaje de error
2	El jefe corrige los datos.	El sistema permite continuar con el proceso

8. Requisito asociado (funcional, no funcional)

Requisito Funcional: RF-02

9. Prototipo de interfaz de usuario

Lista con los distintos Empleados, para seleccionarlo y modificarlo.

Formulario con campos editables: Contraseña y Sueldo

Botones: Actualizar empleado, borrar datos



Actualizar Empleado



Escriba lo que desea actualizar del empleado

Contraseña:

Sueldo:

1850.0

Actualizar empleado

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

1.Nombre del caso de uso del sistema	Borrar película	
2. ID del Caso de uso	CU-08	
3. Descripción del caso de uso		
Un jefe o empleado elimina del sistema una película que ya no debe aparecer en la cartelera. Esto incluye la eliminación de sus datos y la desaparición de sus sesiones si existiesen.		
4. Actor(es)		
Jefe, Empleado		
5. Precondiciones		
<ul style="list-style-type: none">- El actor ha iniciado sesión correctamente.- La película existe en la base de datos.- No hay restricciones que impidan que podamos eliminarla		
6. Postcondiciones		
<ul style="list-style-type: none">- La película ha sido eliminada del sistema.- Las sesiones asociadas a la película también han sido eliminadas		
7. Flujo principal		
<i>[Pasos que describen la realización del caso de uso. Empieza con la primera acción del actor y el sistema emitirá una respuesta]</i>		
Nº	Acción del actor	Respuesta del sistema
1	El actor busca y selecciona la película.	El sistema muestra los detalles de la película.
2	El actor pulsa el botón de "Borrar".	El sistema elimina la película
3	Se confirma la actualización.	El sistema guarda los cambios

7b. Flujo alternativo

[Pasos que describen la realización del caso de uso alternativo]

Nº	Acción del actor	Respuesta del sistema
1	El actor intenta borrar una película sin haberla seleccionado	El sistema advierte al actor que debe seleccionarla
2	El actor selecciona la película y borra	El sistema elimina la película y guarda los cambios

8. Requisito asociado (funcional, no funcional)

Requisito Funcional: RF-06

9. Prototipo de interfaz de usuario

Lista con las distintas Películas, para seleccionarla y eliminarla.

Botón: Borrar película

Nombre: Inception 148 | Correo: CIEN
Nombre: Titanic 195 | Correo: ROMAN
Nombre: Avengers: Endgame 181 | C
Nombre: Coco 105 | Correo: ANIMACI
Nombre: Joker 122 | Correo: DRAMA |
Nombre: Interstellar 169 | Correo: CIE
Nombre: Frozen 102 | Correo: ANIMAC
Nombre: The Matrix 136 | Correo: ACC
Nombre: Up 96 | Correo: ANIMACION
Nombre: La La Land 128 | Correo: MU
Nombre: Joder 120 | Correo: ACCION
Nombre: Maria 120 | Correo: COMEDI

Añadir película

Actualizar película

Borrar película

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

1.Nombre del caso de uso del sistema		Registrarse
3. ID del Caso de uso		CU-15
3. Descripción del caso de uso		
Una persona(próximo cliente) accede al sistema para crear una cuenta proporcionando sus datos personales. El sistema valida los datos y, si todo es correcto, crea el perfil del cliente.		
4. Actor(es)		
Cliente		
5. Precondiciones		
<ul style="list-style-type: none">- El cliente no debe tener una cuenta registrada previamente con el mismo correo electrónico.- El formulario de registro debe estar disponible.		
6. Postcondiciones		
<ul style="list-style-type: none">- El cliente queda registrado en el sistema y puede iniciar sesión.- Se almacena la información del nuevo cliente en la base de datos.		
7. Flujo principal		
<i>[Pasos que describen la realización del caso de uso. Empieza con la primera acción del actor y el sistema emitirá una respuesta]</i>		
Nº	Acción del actor	Respuesta del sistema
1	El cliente accede al formulario de registro.	El sistema muestra el formulario de registro.
2	El cliente introduce los datos requeridos (nombre, apellidos, correo y contraseña)	El sistema valida los datos ingresados.
3	El cliente envía el formulario.	El sistema crea la cuenta y entramos en la ventana de cliente.

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

7b. Flujo alternativo

[Pasos que describen la realización del caso de uso alternativo]

Nº	Acción del actor	Respuesta del sistema
1	El cliente introduce un correo ya registrado.	El sistema muestra un mensaje de error e impide continuar.
2	El cliente corrige el correo electrónico.	El sistema permite reintentar el registro.

8. Requisito asociado (funcional, no funcional)

Requisito Funcional: RF-09

9. Prototipo de interfaz de usuario

Formulario con campos: Nombre, Apellidos, Email, Contraseña

Botones: Registrarse, Borrar datos

Ventana registro
×

Introduzca los datos para su registro

Nombre:

Apellidos:

Correo:

Contraseña:

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

1.Nombre del caso de uso del sistema	Comprar entrada	
4. ID del Caso de uso	CU-01	
3. Descripción del caso de uso		
Un cliente accede a la cartelera, selecciona una sesión, elige un asiento disponible y realiza la compra de una entrada para dicha sesión.		
4. Actor(es)		
Cliente		
5. Precondiciones		
<ul style="list-style-type: none">- El cliente ha iniciado sesión en la plataforma.- Existen sesiones disponibles en la cartelera.- Hay asientos disponibles para la sesión seleccionada.		
6. Postcondiciones		
<ul style="list-style-type: none">- Se genera una entrada válida para la sesión seleccionada.- El asiento elegido queda marcado como ocupado.- Se actualiza las entradas y la disponibilidad en tiempo real.		
7. Flujo principal		
<i>[Pasos que describen la realización del caso de uso. Empieza con la primera acción del actor y el sistema emitirá una respuesta]</i>		
Nº	Acción del actor	Respuesta del sistema
1	El cliente accede a la cartelera.	El sistema muestra la lista de sesiones disponibles.
2	El cliente selecciona una sesión.	El sistema muestra los asientos disponibles.
3	El cliente selecciona un asiento y procede a la compra.	El sistema valida y confirma la compra, generando la entrada.

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

7b. Flujo alternativo

[Pasos que describen la realización del caso de uso alternativo]

Nº	Acción del actor	Respuesta del sistema
1	El cliente selecciona una sesión no disponible	El sistema muestra un mensaje de error
2	El cliente elige otra sesión disponible	El sistema permite reintentar el registro. El sistema permite continuar con la compra.


8. Requisito asociado (funcional, no funcional)

Requisito Funcional: RF-11, RF-12, RF-13, RF-16

9. Prototipo de interfaz de usuario

Cartelera interactiva con películas, horarios y salas.

Combobox de asientos con selección interactiva.

 Sesiones disponibles


×

Sesiones disponibles

Elija la película por la que desea filtrar:

Películas

Título	Sala	Fecha/Hora	Duración (minutos)	Precio (euros)
Joker	5	23/05/2025 21:00	122	10.0
Interstellar	6	24/05/2025 16:00	169	9.5
Frozen	7	24/05/2025 18:30	102	6.0
Up	9	25/05/2025 17:30	96	7.5
Joder	2	25/05/2025 18:30	120	7.0
La La Land	10	25/05/2025 19:00	128	9.0
The Matrix	8	25/05/2025 20:00	136	8.0

 Comprar entrada

×

Comprar entrada

Película: Up

Precio final: 7.5

Sala: 9

Día y hora: 25/05/2025 17:30

Asientos disponibles: 1

Comprar entrada

¡Por la compra de una entrada recibirás 5 puntos!

Por favor, seleccione la sesión de la que quiere comprar una entrada

Comprar entrada

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

1.Nombre del caso de uso del sistema	Añadir Sesión	
5. ID del Caso de uso	CU-04	
3. Descripción del caso de uso		
Un empleado o jefe crea una nueva sesión de cine asignando una película a una sala y una hora específica. La sesión queda disponible para la compra de entradas.		
4. Actor(es)		
Jefe, Empleado		
5. Precondiciones		
<ul style="list-style-type: none">- El actor ha iniciado sesión.- Existen películas y salas registradas en el sistema.- La sala y horario seleccionados están disponibles		
6. Postcondiciones		
<ul style="list-style-type: none">- Se crea una nueva sesión con la información proporcionada.- La sesión aparece en la cartelera para los clientes.		
7. Flujo principal		
<i>[Pasos que describen la realización del caso de uso. Empieza con la primera acción del actor y el sistema emitirá una respuesta]</i>		
Nº	Acción del actor	Respuesta del sistema
1	El actor accede al sistema	El sistema muestra el formulario de añadir sesión
2	El actor selecciona la película, la sala y la hora.	El sistema valida los datos ingresados, y que haya disponibilidad de la clave principal
3	El actor confirma la creación de la sesión.	El sistema registra la sesión y la agrega a la cartelera.

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

7b. Flujo alternativo

[Pasos que describen la realización del caso de uso alternativo]

Nº	Acción del actor	Respuesta del sistema
1	El actor selecciona una sala ya ocupada en ese horario.	El sistema muestra un mensaje de error y no permite continuar.
2	El actor selecciona otro horario o sala.	El sistema vuelve a validar y permite continuar.

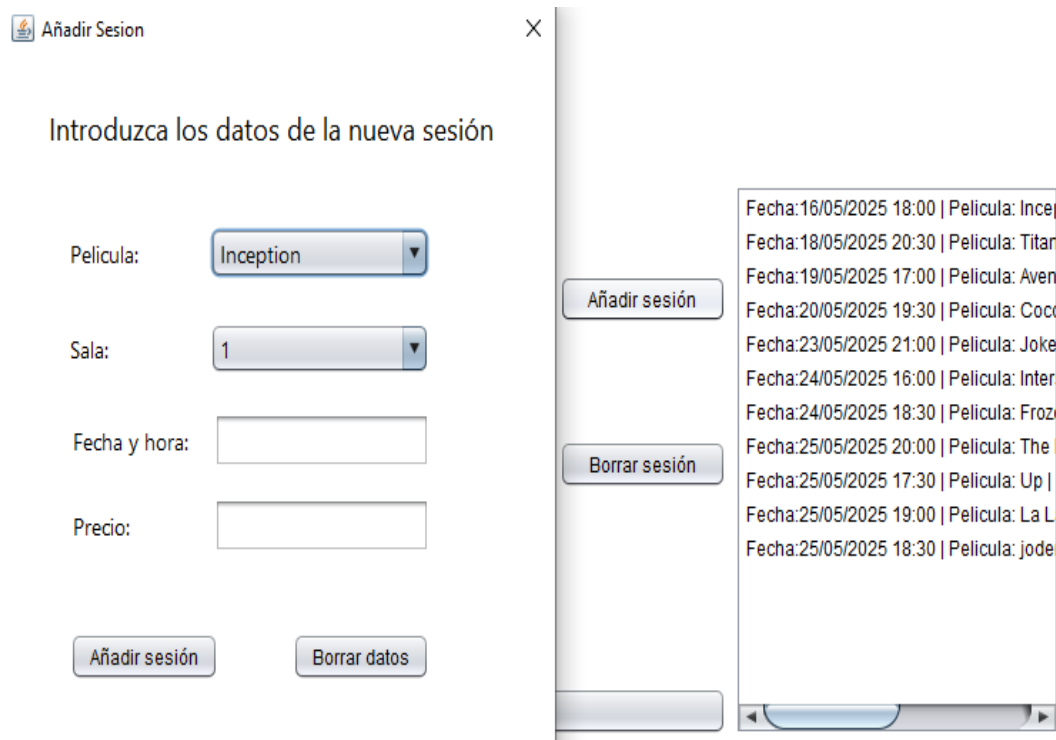
8. Requisito asociado (funcional, no funcional)

Requisito Funcional: RF-07

9. Prototipo de interfaz de usuario

Formulario con campos: Película, Sala, Horario y Precio

Botones: Añadir Sesión, borrar datos



	<p align="center">CineNova</p> <p align="center">Especificación de Requisitos</p>	<p align="center">IES.CAMAS</p>
---	---	--

5.2 Requisitos Funcionales

- RF01 - El sistema debe permitir a los jefes añadir nuevos empleados al sistema.
- RF02 - El sistema debe permitir a los jefes modificar los datos de empleados existentes.
- RF03 - El sistema debe permitir a los jefes eliminar empleados del sistema.
- RF04 - El sistema debe permitir a empleados y jefes añadir nuevas películas a la cartelera.
- RF05 - El sistema debe permitir a empleados y jefes modificar la información de las películas (título, duración, sinopsis, clasificación, etc.).
- RF06 - El sistema debe permitir a empleados y jefes eliminar películas de la cartelera.
- RF07 - El sistema debe permitir a empleados y jefes crear sesiones asignando una película a una sala y hora específica.
- RF08 - El sistema debe permitir a empleados y jefes eliminar sesiones programadas.
- RF09 - El sistema debe permitir a los clientes registrarse en la plataforma.
- RF10 - El sistema debe permitir a los clientes iniciar sesión mediante autenticación con sus credenciales.
- RF11 - El sistema debe mostrar a los clientes una lista actualizada de sesiones disponibles, incluyendo película, sala, horario y disponibilidad de asientos.
- RF12 - El sistema debe permitir a los clientes seleccionar una sesión, elegir un asiento concreto y comprar una entrada.

	<p align="center">CineNova</p> <p align="center">Especificación de Requisitos</p>	<p align="center">IES.CAMAS</p>
---	---	--

- RF13 - El sistema debe generar un recibo tras la compra de una entrada.
- RF14 - El sistema debe permitir a los clientes descargar el recibo de la entrada en formato digital (PDF u otro).
- RF15 - El sistema debe permitir a los clientes cancelar una entrada comprada,
- RF16 - El sistema debe actualizar la disponibilidad de asientos en tiempo real tras cada compra o cancelación.

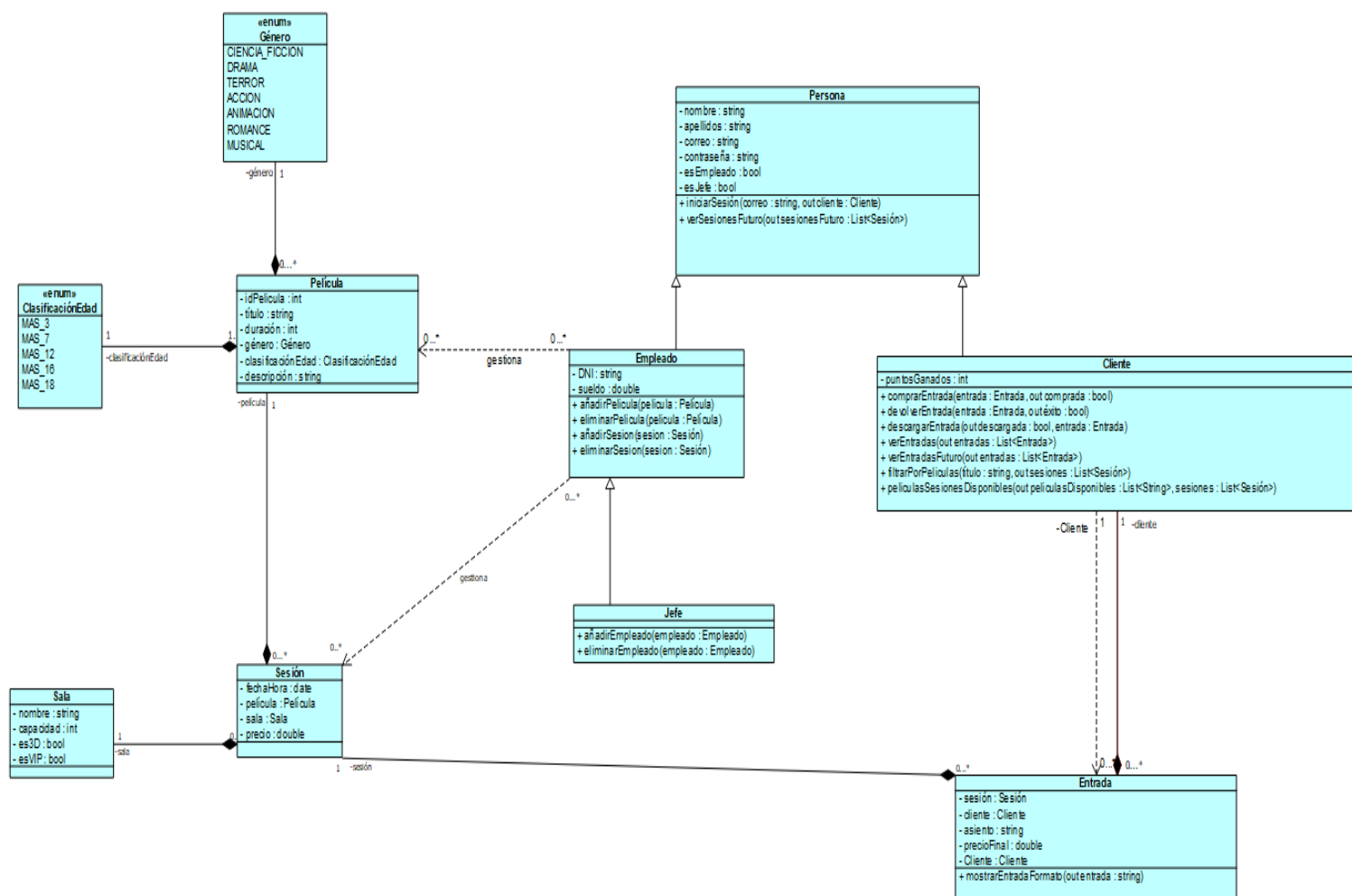
5.3 Requisitos No Funcionales

- RNF01 - El sistema debe tener una interfaz intuitiva y amigable tanto para empleados como para clientes, reduciendo la necesidad de formación previa.
- RNF02 - El sistema debe mostrar mensajes claros y descriptivos en caso de errores o acciones fallidas
- RNF03 - El sistema debe validar toda la información introducida por el usuario para prevenir ataques como inyecciones SQL.
- RNF04 - El acceso a funcionalidades debe estar controlado por roles: los jefes podrán acceder a más funcionalidades que los empleados, y estos más que los clientes.
- RNF05 - El sistema debe ser compatible con bases de datos Oracle para facilitar la gestión de los datos.
- RNF06 - El código fuente debe estar documentado adecuadamente para facilitar el mantenimiento por parte de otros desarrolladores.

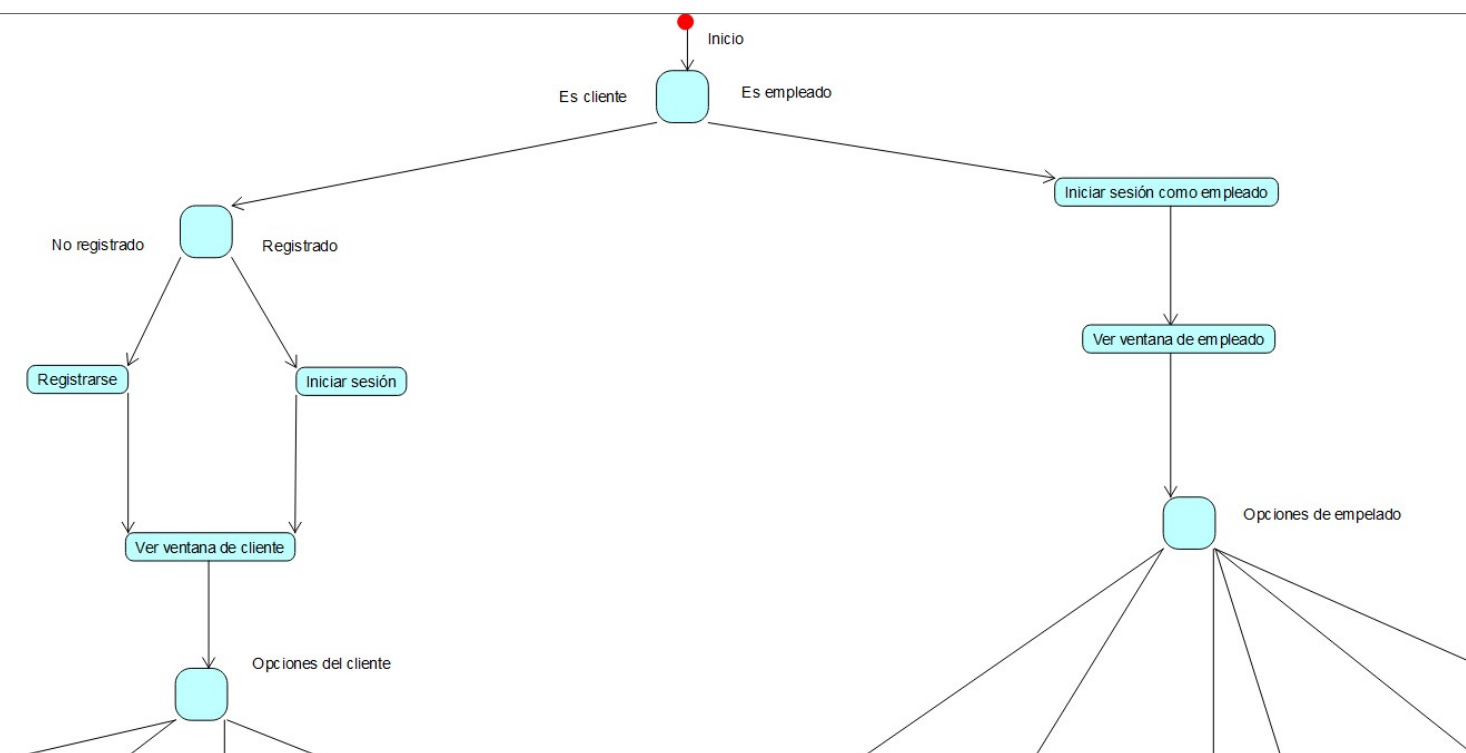
 CineNova	<p style="text-align: center;">CineNova</p> <p style="text-align: center;">Especificación de Requisitos</p>	<p style="text-align: center;">IES.CAMAS</p>
--	---	---

6 Modelado del Sistema (Diseño)

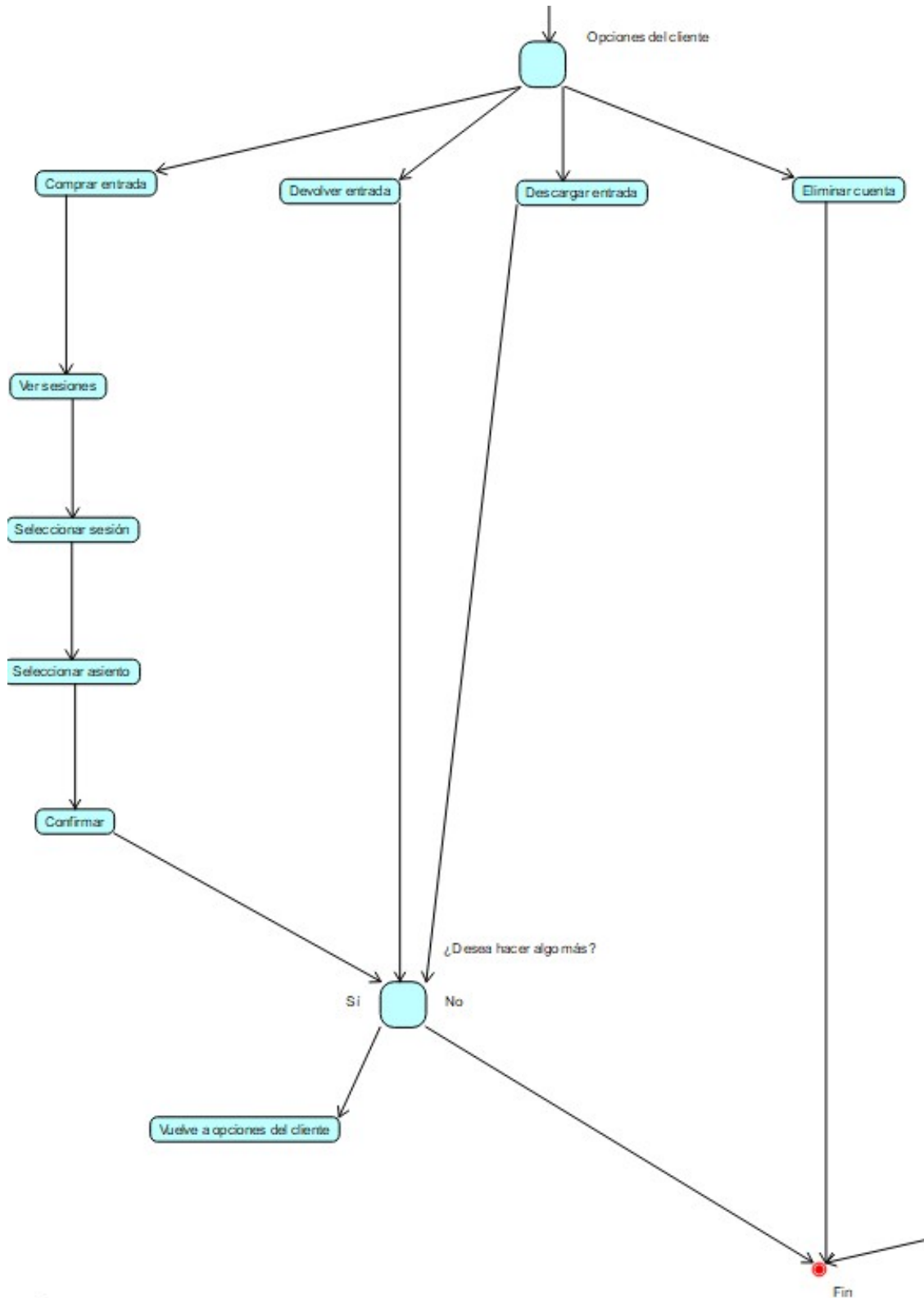
6.1 Diagrama de clases

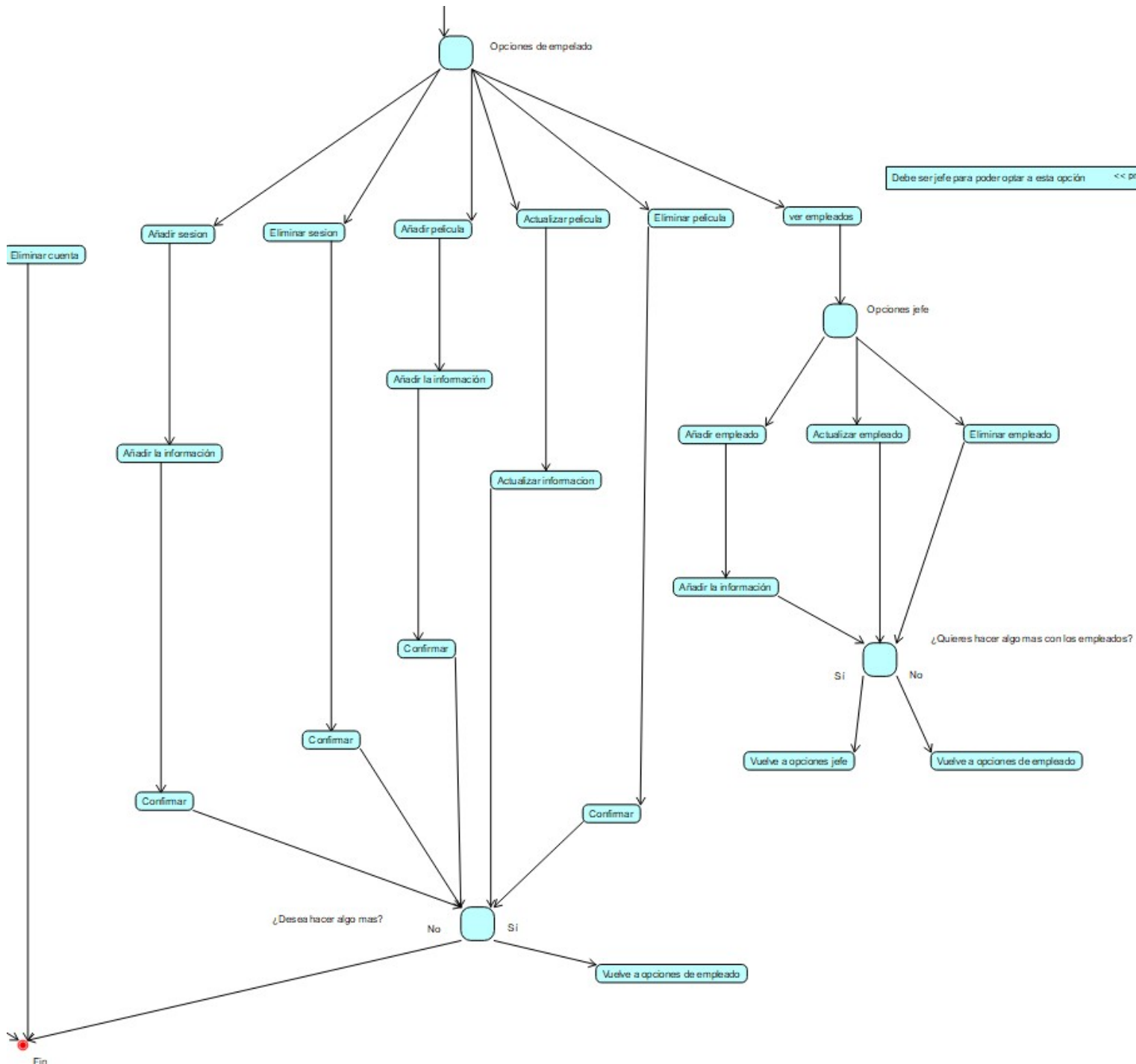


6.3 Diagrama de Actividad



A continuación vamos a dividir el diagrama de actividad en la parte de empleado y la parte de cliente mostradas en la captura de arriba para una mejor legibilidad.





	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

7 Plan de pruebas y validación

7.1 Plan de Pruebas Unitarias

A continuación, se describen las pruebas unitarias que hemos diseñado para verificar el correcto funcionamiento de cada unidad funcional del sistema. Estas pruebas se desarrollan con JUnit y se gestionan a través de Maven en un entorno de control de versiones con GitHub.

Cada prueba unitaria corresponde con uno o más requisitos funcionales de los detallados en el punto anterior.

Hemos hecho pruebas unitarias por cada Clase java que tenemos en nuestro proyecto, algunos ejemplos para entender el 7.2 y su funcionamiento pueden ser:

Prueba unitaria	Descripción	Resultado esperado
testAñadirEmpleado()	Verifica que el jefe puede añadir un nuevo empleado con datos válidos.	El empleado se inserta correctamente en la base de datos.
testActualizarPelicula()	Comprueba que se puede modificar la información de una película existente.	Los cambios son guardados y visibles al consultarlos.
testBorrarSesion()	Comprueba que una sesión existente puede ser eliminada.	La sesión es eliminada correctamente de la base de datos.
testRegistrarClienteÉxito()	Verifica que un nuevo cliente puede registrarse correctamente.	El cliente es registrado y almacenado en la base de datos.
testPeliculasSesionesDisponibles()	Comprueba que se muestra correctamente la cartelera y disponibilidad.	Se devuelve una lista actualizada de sesiones disponibles.
testEntradaDescargada()	Verifica que un cliente puede descargar su recibo en formato digital.	Se genera un archivo descargable con los datos de la compra.

	CineNova Especificación de Requisitos	IES.CAMAS
---	--	------------------

7.2 Trazabilidad de Requisitos

Relaciona requisitos con sus pruebas para garantizar que todos han sido cubiertos.

RF-ID	Descripción del Requisito Funcional	Caso de Uso Relacionado	Clases Implicadas	Prueba Unitaria	Estado
RF-01	El jefe podrá añadir nuevos empleados al sistema	CU-11: Añadir empleado	Jefe, Empleado	testAñadirEmpleado()	Implementado
RF-02	El sistema debe permitir a los jefes modificar los datos de empleados existentes.	CU-13: Actualizar empleado	Jefe, Empleado	testActualizarEmpleado()	Implementado
RF-03	El sistema debe permitir a los jefes eliminar empleados del sistema.	CU-12: Borrar empleado	Jefe, Empleado	testBorrarEmpleado()	Implementado
RF-04	El sistema debe permitir a empleados y jefes añadir nuevas películas a la cartelera.	CU-07: Añadir película	Jefe, Empleado, Película	testAñadirPelícula()	Implementado
RF-05	El sistema debe permitir a empleados y jefes modificar la información de las películas (título, duración, sinopsis, clasificación, etc.).	CU-09: Actualizar película	Jefe, Empleado, Película	testActualizarPelícula()	Implementado
RF-06	El sistema debe permitir a empleados y jefes eliminar películas de la cartelera.	CU-08: Borrar película	Jefe, Empleado, Película	testBorrarPelícula()	Implementado
RF-07	El sistema debe permitir a empleados y jefes crear sesiones asignando una película a una sala y hora específica.	CU-04: Añadir Sesión	Jefe, Empleado, Sesión, Película, Sala	testAñadirSesión()	En desarrollo
RF-08	El sistema debe permitir a empleados y jefes eliminar sesiones programadas.	CU-05: Borrar sesión	Jefe, Empleado, Sesión, Película, Sala	testBorrarSesión()	En desarrollo
RF-09	El sistema debe permitir a los clientes registrarse en la plataforma.	CU-15: Registrarse	Cliente	testRegistrarClienteÉxito()	Implementado
RF-10	El sistema debe permitir a los clientes iniciar sesión mediante autenticación con	CU-16: Iniciar sesión	Cliente	testIniciarSesión()	Implementado

 CineNova	CineNova Especificación de Requisitos	IES.CAMAS
--	--	------------------

	sus credenciales.				
RF-11	El sistema debe mostrar a los clientes una lista actualizada de sesiones disponibles, incluyendo película, sala, horario y disponibilidad de asientos.	CU-01: Comprar entrada	Cliente, Entrada, Sesión, Película, Sala	testPelículasSesionesDisponibles()	Implementado
RF-12	El sistema debe permitir a los clientes seleccionar una sesión, elegir un asiento concreto y comprar una entrada.	CU-01: Comprar Entrada	Cliente, Entrada, Sesión, Película, Sala	testPelículasSesionesDisponibles()	Implementado
RF-13	El sistema debe generar un recibo tras la compra de una entrada.	CU-03: Comprar entrada	Cliente, Entrada	testEntradaComprada()	Implementado
RF-14	El sistema debe permitir a los clientes descargar el recibo de la entrada en formato digital	CU-03: Descargar entrada	Cliente, Entrada	testEntradaDescargada()	Implementado
RF-15	El sistema debe permitir a los clientes cancelar una entrada comprada	CU-02: Devolver entrada	Cliente, Entrada	testEntradaDevuelta()	Implementado
RF-16	El sistema debe actualizar la disponibilidad de asientos en tiempo real tras cada compra o cancelación.	CU-01: Comprar entrada	Cliente, Entrada, Sesión, Película, Sala	testPelículasSesionesDisponibles()	Implementado

	<p style="text-align: center;">CineNova Especificación de Requisitos</p>	<p style="text-align: center;">IES.CAMAS</p>
---	--	---

8 Gestión configuración

8.1 Estructura del repositorio

Utilizamos para el control de versiones la herramienta Git, más concretamente GitHub. En ella hemos subido los diferentes cambios a lo largo del desarrollo del proyecto a través de commits a un repositorio donde hemos trabajado en conjunto.

Dicho repositorio se estructura de la siguiente manera. Existen 5 ramas y estas se dividen en 4 categorías:

Rama principal

- Rama **main**: En esta rama se encuentra el proyecto al completo con todas sus partes. También en esta rama se han llevado a cabo los cambios relacionados con la documentación del proyecto y el archivo sql que define la base de datos en las carpetas docs y sql respectivamente. Dentro de la carpeta docs se encuentra la memoria del proyecto en la cual se recoge la especificación de requisitos, la documentación javadoc del código y el informe de cobertura de pruebas realizado con jacoco.

Ramas de desarrollo

- Rama **desarrollo_cliente**: En esta rama se ha llevado a cabo la implementación de la parte relacionada con el cliente y su interacción con la aplicación. Esta implementación ha sido realizada con el lenguaje de programación Java.
- Rama **desarrollo_empleado**: En esta rama se ha llevado a cabo la implementación de la parte relacionada con los empleados y los jefes y su interacción con la aplicación. Esta implementación ha sido realizada con el lenguaje de programación Java.

Rama de diseño

- Rama **modelado**: En esta rama se ha llevado a cabo la realización de los diferentes diagramas de diseño para poder entender y estructurar como funciona la aplicación. Para realizar dichos diagramas hemos usado Umbrello.

	<p style="text-align: center;">CineNova Especificación de Requisitos</p>	<p style="text-align: center;">IES.CAMAS</p>
---	--	---

Rama de pruebas

- Rama **pruebas**: En esta rama se han llevado a cabo las diferentes pruebas unitarias realizadas con Junit 5 y la medición de cobertura de dichas pruebas usando jacoco.

8.2 Gestor del Ciclo de Vida

El proyecto esta configurado con Maven para facilitar la implementación y mantenimiento. Maven permite a través de un archivo llamado pom.xml gestionar dependencias y facilita la realización y comprobación de pruebas o tests.

Entre las dependencias que hemos añadido al proyecto están la **dependencia de JDBC** para poder conectar el código Java con la base de datos de Oracle, la **dependencia de Jacoco** para poder tener un informe sobre las cobertura de pruebas unitarias y asi tener una idea general de lo que se ha abarcado con ellas y por último una **dependencia que nos permite generar el archivo jar** (ejecutable) con las dependencias ya comentadas.

Con Maven podemos ejecutar diferentes comandos en la línea de comandos siempre que estemos ubicados en la ruta del proyecto. Estos comandos nos ayudan a poder realizar diferentes acciones que vamos a explicar a continuación:

- **mvn clean**: usando esta orden podremos limpiar el directorio target que se genera con los archivos de salida cuando se hace un build del proyecto.
- **mvn install**: esta orden nos permite construir el proyecto para asi poder generar los archivos en la carpeta target, entre ellos el archivo jar o ejecutable para poder usar la aplicación.
- **mvn test**: esta orden nos permite realizar los tests o pruebas sin tener que construir el proyecto.
- **mvn deploy**: esta orden nos permite desplegar el proyecto. En nuestro caso no es útil ya que el proyecto por el momento se está realizando de forma local pero si en el futuro se escalara y se subiera a un servidor habría que usar esta orden para ello.

Por último comentar con para construir el proyecto se puede hacer **mvn build** o utilizar los comandos mvn clean y mvn install uno detras de otro para conseguir el mismo resultado. Para generar el javadoc habrá que hacerlo desde el IDE.