Pestaña 1





**Curso: 5°8**

**Ciclo lectivo: 2025**

**1er Cuatrimestre**

**Materia: Proyecto Informatico II**

**Profesor: Alan Edmundo Etkin**

**Integrantes: Laura Martínez, Elmer Flores Arce, Ezequiel Toriaci, Andrew Talavera y Leandro Vilchez.**

******



**Proyecto Cine: informe del segundo sprint**

### **Estudio de requerimientos**

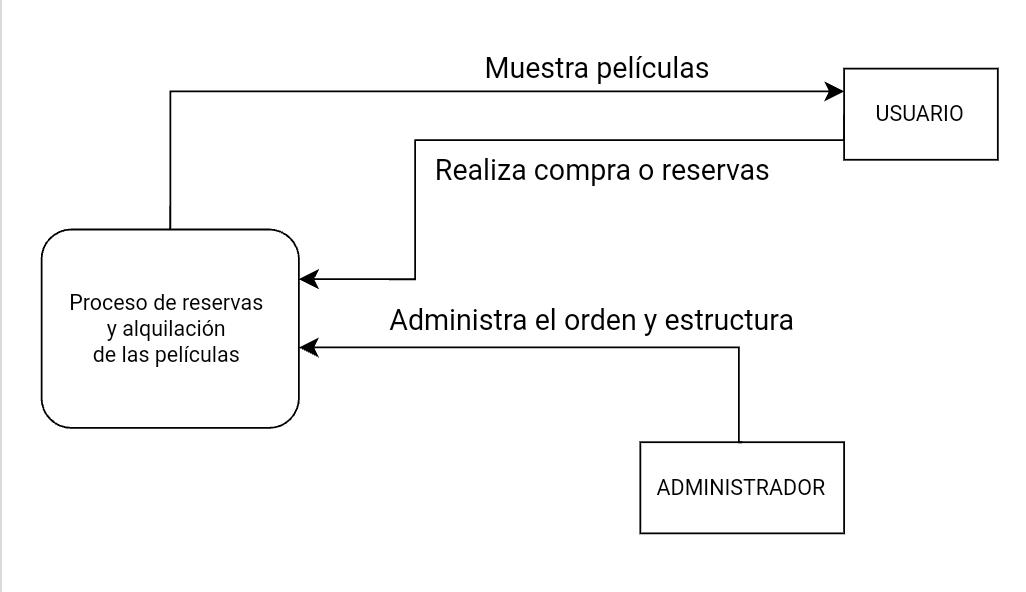
#### Reconocimiento

Para tener un punto claro de lo que queremos realizar en el desarrollo de nuestro proyecto empezar, analizamos diferentes sistemas de venta y administración de entradas de cines reconocidos, como los de Cinemark Hoyts, Cinépolis, y Village Cines.

Esos sistemas de cine permiten a los usuarios:

1. Comprar entradas de manera online o presencial.
2. Seleccionar la función (película, fecha, horario).
3. Elegir el asiento específico dentro de una sala dividida en sectores, filas y butacas.
4. Recibir un ticket digital con código QR o código de barras para validar el ingreso.
5. Obtener una factura electrónica a la compra.

Utilizando las herramientas de análisis como el diagrama de contexto, comenzamos a definir la mejor solución que se adapte al sistema de venta y administración de entradas para el cine.



Después de analizar toda la información recolectada, optamos por realizar un sistema similar a los ya existentes, pero adaptado a las necesidades específicas del proyecto:

Un cine que ofrezca una venta online de entradas con un proceso de compra confiable

***Propuesta***

***Después de haber realizado diversos estudios (tales como el estudio de mercado) y análisis (FODA ,reconocimiento ,relevamiento , etc), se resumen los requerimientos del sistema que son:***

La posibilidad de que el usuario pueda comprar o alquilar la película mediante métodos de pago implementados es este sistema, también que pueda registrarse para poder tener guardadas las películas que haya alquilado. Todo esto con una interfaz atractiva e interactiva, dentro de una página web hecha para que sea usada desde dispositivos móviles o computadoras. Además, el sistema deberá permitir la visualización de las salas de cine y la selección de butacas en tiempo real, mostrando cuales están disponibles y cuales ya fueron reservadas, para que el usuario pueda elegir la que más prefiera.

Una vez realizada la compra, se generará un ticket digital que contará con un código único para poder ingresar a la función.

Ahora desde el lado de los administradores del cine, tendrán acceso a una interfaz para poder gestionar las funciones disponibles, asignar horarios, ver reportes de ventas, administrar las películas que se encuentran para alquilar, y gestionar los usuarios registrados.

También, se tendrá en cuenta la generación de una factura digital en cada compra, cumpliendo con los requisitos legales correspondientes de Argentina.

**Estudio Factibilidad**

### **Factibilidad técnica**

Durante el primer sprint, se propuso una arquitectura basada en PHP y MySQL. Sin embargo, tras hacer un análisis más profundo y los avances en el diseño técnico, se decidió optar por una arquitectura basada en Django (Python) para el backend y HTML y CSS para el frontend, con una posibilidad de utilizar JavaScript.

También se mantuvo el uso de MySQL como motor de base de datos, lo cual se adapta perfectamente a Django y permite trabajar con estructuras, como el manejo de usuarios, funciones, salas y butacas. A nivel de modelado, se avanzó con diagramas entidad-relación, casos de uso y diagramas de secuencia que ayudaron a clarificar el flujo del sistema y las relaciones entre componentes.

### **De recursos humanos:**

En este segundo sprint se reforzaron los roles dentro del equipo. Los integrantes ya no solo comparten tareas, sino que comienzan a especializarse: uno se enfoca en el backend con Django, otros en el diseño visual y el frontend, y un miembro gestiona pruebas y documentación. Esta asignación mejora la eficiencia y permite un avance más claro en el desarrollo del sistema.

El team leader continúa organizando y facilitando la comunicación, mientras que las tareas se comparten en base a la planificación del Gantt. Se incorporó el uso de herramientas colaborativas como Google Drive y GitHub, lo que facilita el trabajo remoto y el control de versiones.

En resumen, el equipo actual demostró tener la capacidad de adaptarse a nuevas herramientas y tecnologías según lo que necesite el diseño.

### **Factibilidad económica**

Con el avance del segundo sprint y después de definir cómo va a estar hecho el sistema, decidimos ajustar el presupuesto. Esto se debe a que cambiaron algunos precios y también la cantidad de trabajo necesaria en cada etapa.

A diferencia de lo que pensábamos al principio, ahora el proyecto necesita más tiempo dedicado a la parte del backend, el uso de herramientas para controlar versiones del código, el diseño de pantallas y pruebas técnicas. Por eso, repartimos de nuevo las horas de trabajo según cada rol y actualizamos los costos. También tuvimos que modificar los valores de algunos materiales, como las notebooks, el router, etc.

A continuación, se presenta la estimación de costos actualizada por sprint, incluyendo los recursos físicos y digitales necesarios para completar el sistema.

#### **Sprint 1 –** Reconocimiento y relevamiento

Este sprint se enfocó en la definición general del sistema, la investigación de mercado, el FODA, la formulación del OLA y las entrevistas. Se estimaron 44 horas de trabajo, con la siguiente distribución de roles:

* Team Leader: 8 h × $31.000 = $248.000
* Analistas: 16 h × $27.500 = $440.000
* Diseñador: 10 h × $22.500 = $225.000
* Programador: 5 h × $26.000 = $130.000
* Tester: 5 h × $25.500 = $127.500

**Subtotal Sprint 1 (Mano de obra):** $1.170.500  
**Equipamiento físico y servicios asignados al sprint 1:** $1.219.504 (del presupuesto anterior)  
**Total Sprint 1:** $2.390.004

#### **Sprint 2 –** Modelado, arquitectura y diseño

Este sprint abarcó la elección del entorno de desarrollo, el modelado del sistema (casos de uso, diagramas de secuencia, entidad-relación), y el diseño de interfaces. Se distribuyeron 60 horas de trabajo:

* Team Leader: 10 h × $31.000 = $310.000
* Programador: 20 h × $26.000 = $520.000
* Diseñador: 15 h × $22.500 = $337.500
* Tester: 10 h × $25.500 = $255.000
* Analistas: 5 h × $27.500 = $137.500

**Subtotal Sprint 2 (Mano de obra):** $1.560.000  
**Servicios asignados a este sprint:** Hosting ($9.500), Dominio ($8.700), Base de datos ($4.200), certificado SSL ($62.000)  
**Total servicios:** $84.400  
**Total Sprint 2:** $1.644.400

#### **Sprint 3 – Implementación, pruebas y ajustes finales**

Este sprint estima la programación completa, la integración del sistema, las pruebas funcionales, correcciones, y la preparación del prototipo ejecutable. Se asignaron 80 horas de trabajo:

* Programador: 35 h × $26.000 = $910.000
* Tester: 20 h × $25.500 = $510.000
* Team Leader: 10 h × $31.000 = $310.000
* Diseñador: 10 h × $22.500 = $225.000
* Analistas: 5 h × $27.500 = $137.500

**Total Sprint 3 (Mano de obra):** $2.092.500No se agregan nuevos servicios ni materiales en esta etapa.

#### **Equipamiento físico actualizado**

Basado en los nuevos valores informados:

* Notebooks: $395.000 × 4 = $1.580.000
* Impresoras: $26.000 × 4 = $104.000
* Instalación de red: $2.500.000
* Router: $45.000

**Total equipamiento físico:** **$4.229.000**

#### **Resumen general del proyecto**

* **Total Sprint 1:** $2.390.004
* **Total Sprint 2:** $1.644.400
* **Total Sprint 3:** $2.092.500

**TOTAL DE LOS 3 SPRINTS:** $6.126.904

**Total mano de obra (3 sprints):** $4.722.904  
**Total servicios web y software:** $84.400  
**Total equipamiento físico:** $4.229.000

Presupuesto del Proyecto

# **1. Equipamiento Físico (Hardware)**

| Elemento | Precio Unitario | Cantidad | Total (ARS) |
| --- | --- | --- | --- |
| Notebooks | $395,000.00 | 4 | $1,580,000.00 |
| Impresoras | $26,000.00 | 4 | $104,000.00 |
| Instalación de Red | $2,500,000.00 | 1 | $2,500,000.00 |
| Router | $45,000.00 | 1 | $45,000.00 |

Total Equipamiento: $4,229,000.00

# **2. Servicios Web y Software**

| Elemento | Precio Unitario | Cantidad | Total (ARS) |
| --- | --- | --- | --- |
| Hosting Web (anual) | $9,500.00 | 1 | $9,500.00 |
| Dominio de Internet | $8,700.00 | 1 | $8,700.00 |
| Base de Datos (6 meses) | $4,200.00 | 1 | $4,200.00 |
| Certificado SSL | $62,000.00 | 1 | $62,000.00 |

Total Servicios Web y Software: $84,400.00

# **3. Costos por Hora de Cada Rol**

| Miembro | Precio por Hora |
| --- | --- |
| Team Leader | $31,000 |
| Analistas | $27,500 |
| Programador | $26,000 |
| Diseñador | $22,500 |
| Tester | $25,500 |

# **4. Etapas de Desarrollo**

| Etapa | Actividades Principales | Horas Trabajadas | Costo Total |
| --- | --- | --- | --- |
| 1 | Reconocimiento y relevamiento | 44 | $1,219,504 |
| 2 | Estudio del mercado, FODA, OLA | 45 | $1.222.500 |
| 3 | Diagrama de contexto | 50 | $1.377.500 |
| 4 | Programación y testing inicial | 65 | $1.737.500 |
| 5 | Ajustes finales | 30 | $782.500 |

Nota: Los costos de las siguientes etapas se agregan a medida que avance el desarrollo.

# **5. Resumen General**

| Concepto | Total (ARS) |
| --- | --- |
| Equipamiento Físico | $4,229,000.00 |
| Servicios Web y Software | $84,400.00 |
| Costo de Equipo (Etapa 1) | $1,219,504.00 |

TOTAL ESTIMADO ACTUAL: $5,532,904.00

***Objetivos, límites y alcance***

**OLA**

**OBJETIVOS:**

Los objetivos del desarrollo son:

* Crear un sistema informático para :

1. La compra/venta y alquiler de películas
2. Que el usuario pueda visualizar la película que alquiló
3. Poder administrar los usuarios registrados
4. Lograr una administración de las compras y ventas

**LÍMITE:**

Límites de nuestro proyecto cine:

* Una funcionalidad que se podría hacer en nuestro sistema, sería el poder pedir tu combo junto a tu entrada, para que después lo canjee en el local físico. Ese seria el limite de nuestro proyecto, por temas de tiempo y lógica no se pudo agregar
* Otra funcionalidad que podría existir, serian los descuentos segun el dia de semana o si su condición física, pero de coordinación con el cliente no se pudo llegar a contemplar

**ALCANCE:**

* El sistema permitirá a los usuarios registrarse, comprar, vender y alquilar películas de manera digital, así como visualizar las películas que hayan alquilado desde su cuenta personal. También se incorporarán funciones administrativas para gestionar usuarios y controlar las operaciones de compra y venta.

Sin embargo, debido a limitaciones de tiempo y complejidad, no se incluirán funcionalidades como la compra de combos junto con la entrada, ni descuentos especiales según el día o la condición física del usuario. Estas características se reconocen como posibles mejoras futuras, pero quedan fuera del desarrollo actual.

**Diseño del modelo y arquitectura**

#### **Arquitectura**

**Decisiones de la arquitectura del sistema**

Para el diseño de la arquitectura del sistema se decidió utilizar el modelo MVC (Modelo-Vista-Controlador). Esta elección responde a las necesidades específicas del proyecto, que consiste en una aplicación web para un cine, en la cual los usuarios pueden consultar la cartelera, ver detalles de las películas, seleccionar horarios y realizar reservas o compras de entradas.

**Justificación de la elección del modelo MVC**

El patrón MVC divide la aplicación en tres componentes principales:

Modelo: se encarga de gestionar la información del sistema, como películas, horarios, salas, asientos disponibles, usuarios registrados y reservas.

Vista: presenta la información al usuario final. En este caso, serían las páginas web donde se muestra la cartelera, los formularios de reserva y las confirmaciones de compra.

Controlador: gestiona las acciones del usuario, como seleccionar una película, elegir una función y confirmar una reserva. Conecta los datos del modelo con las vistas correspondientes.

Esta estructura permite mantener una separación clara entre la lógica del negocio (reserva de entradas, disponibilidad de funciones) y la interfaz con la que interactúa el usuario. Esto facilita el desarrollo, mantenimiento y escalabilidad del sistema.

**Ventajas del modelo aplicado**

Facilita la modificación del diseño del sitio web sin afectar la lógica de funcionamiento (por ejemplo, cambiar el estilo de la cartelera sin alterar el sistema de reservas).

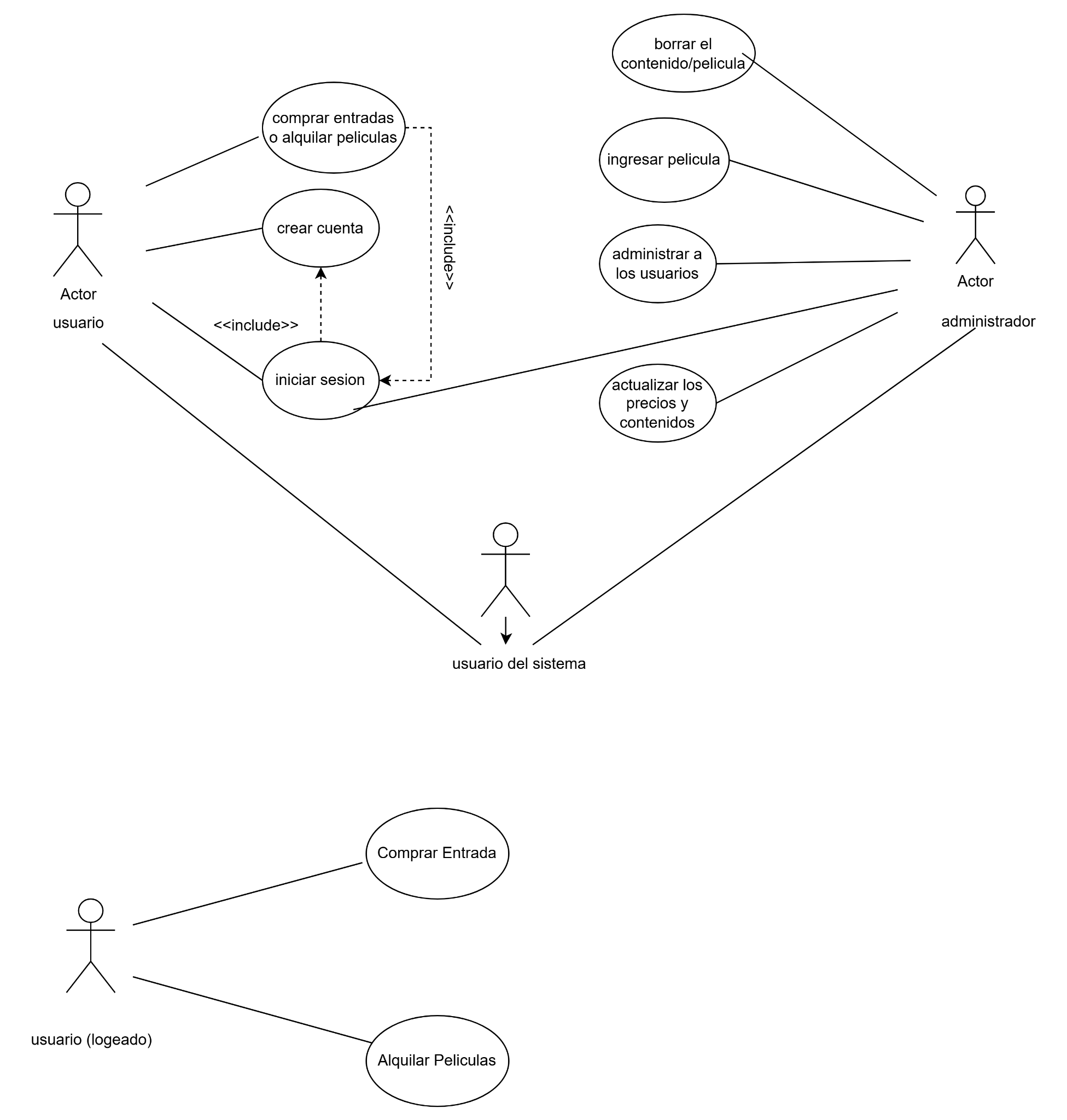
Permite reutilizar componentes del sistema, como el modelo de películas o usuarios, en distintas vistas (página principal, buscador, entre otros.).

Favorece el trabajo colaborativo, ya que diseñadores, programadores y desarrolladores back-end pueden trabajar en paralelo.

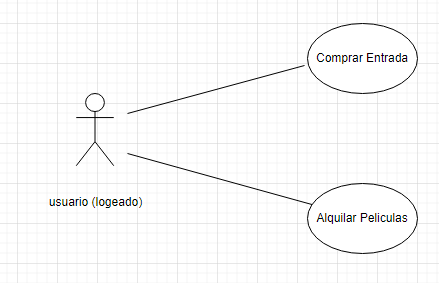
Mejora la organización y claridad del código, lo que reduce errores y simplifica futuras ampliaciones, como agregar nuevas funcionalidades (trailers, valoraciones, etc.).

#### **Modelo**

***Casos de uso***

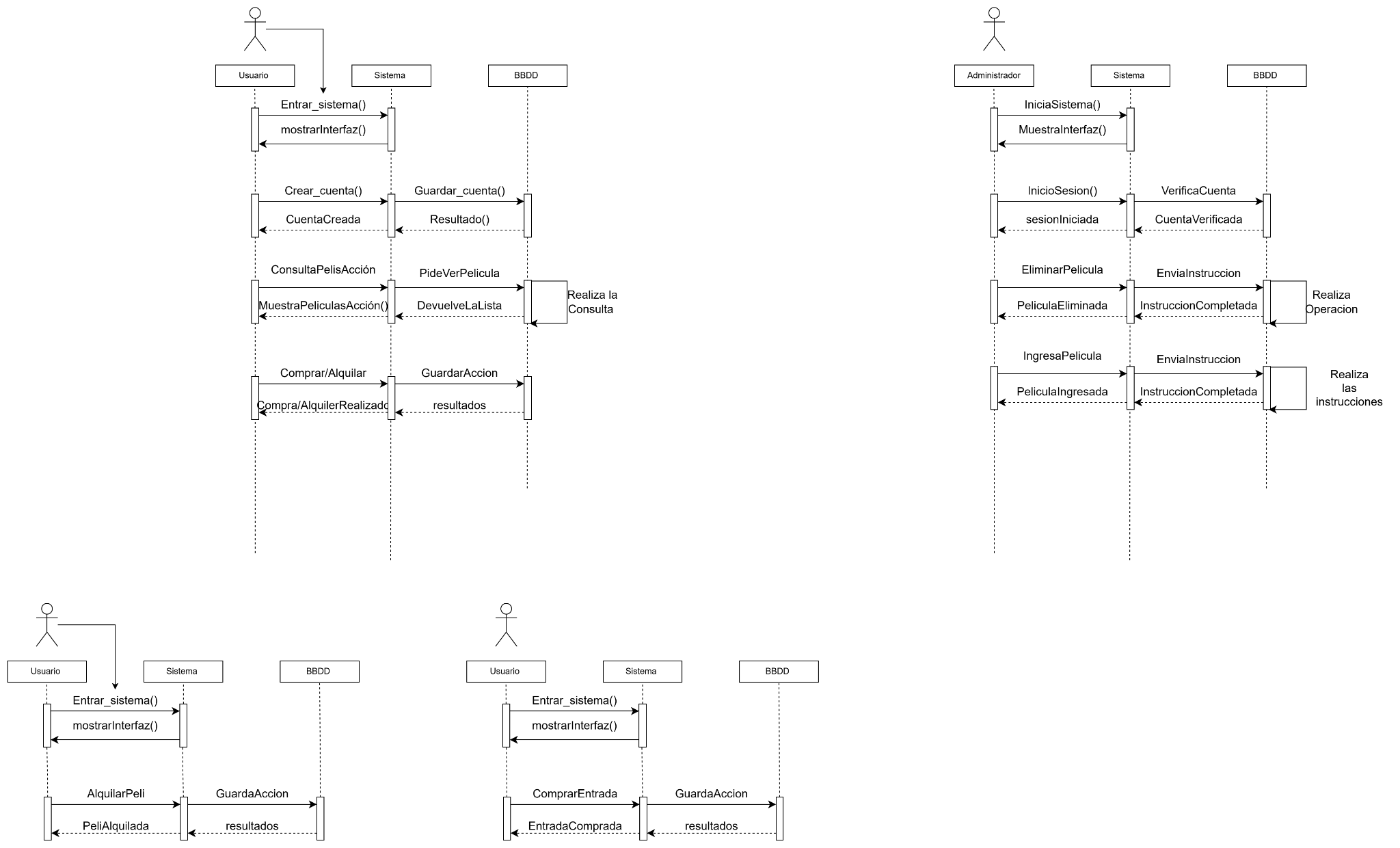
******

***caso de uso detallado-usuario realiza acción***

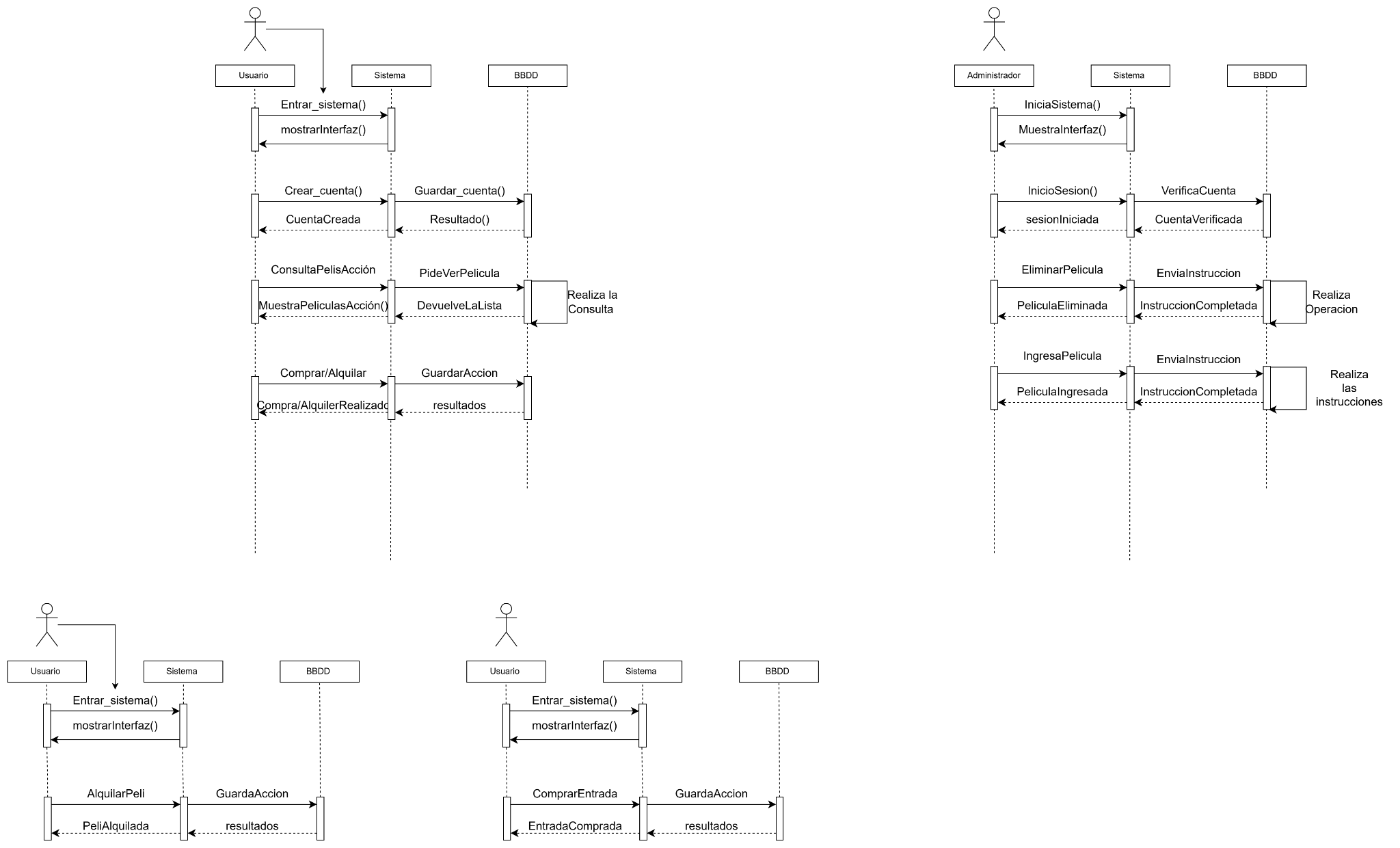


**Diagrama de Secuencia**

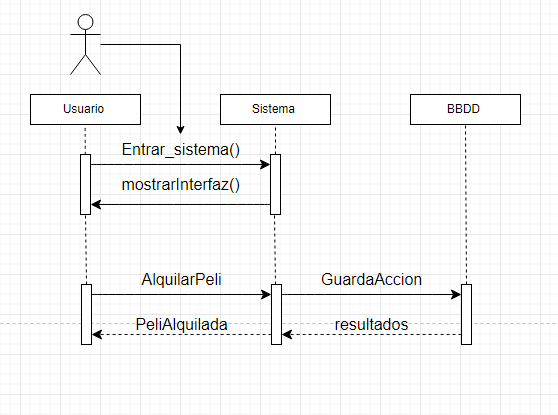
***Usuario-nivel general***



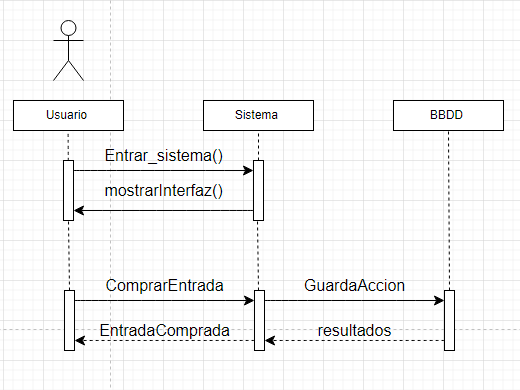
***Administrador-nivel general***



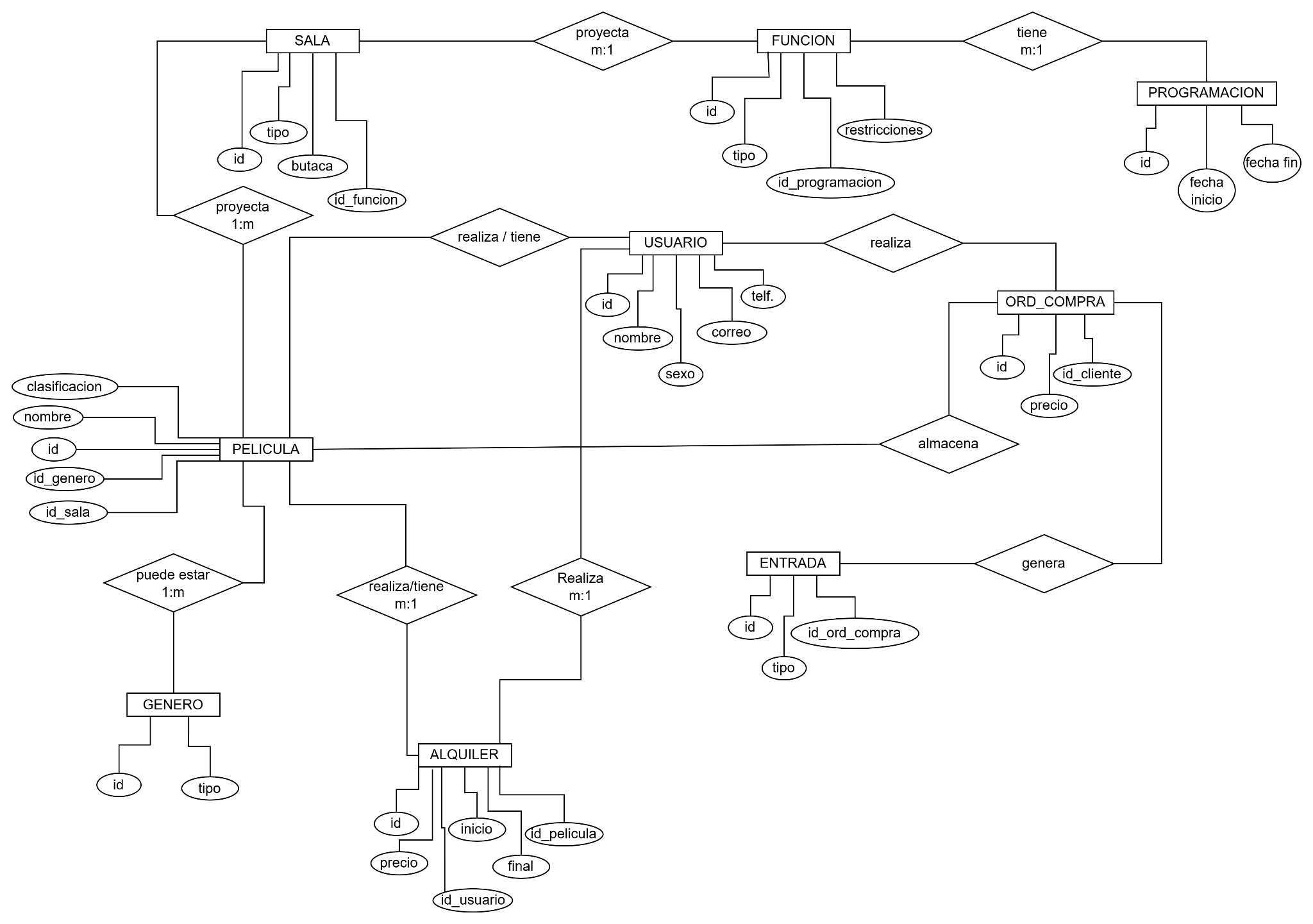
***Acción de alquilar-usuario detallado***



***acción de compra-usuario detallado***



***Diagrama Entidad-Relación***



#### **Entorno de desarrollo**

IDEs : Visual Studio Code

Visual Studio Code es un entorno de desarrollo gratuito de código fuente multiplataforma y gratuito desarrollado por Microsoft. Es una herramienta popular para programadores con soporte para Python, HTML, css y otras tecnologías que serán necesarias para más adelante.

Motivo de elección

Es de código abierto , además de contener muchas extensiones útiles para usar,

Marco de desarrollo: Django

Django es un framework web de alto nivel, gratuito y de código abierto, escrito en Python. Se utiliza para desarrollar aplicaciones web de forma rápida y eficiente, simplificando la creación de aplicaciones web backend.

Motivo de elección:

Para facilitar y acelerar el proceso de creación de aplicaciones web

Lenguaje de programación: Python

Python es un lenguaje de programación de alto nivel, interpretado, interactivo y orientado a objetos. Es conocido por su legibilidad y simplicidad, lo que lo hace ideal para usar. Se utiliza en una amplia variedad de aplicaciones, desde desarrollo web y software

Motivo de elección:

Ya tenemos conocimientos y además de tener un amplio soporte de bibliotecas (Django)

Motores de bases de datos: MySQL

MySQL Workbench es una interfaz gráfica de usuario (GUI) o herramienta de gestión gráfica que nos permite interactuar con la base de datos de MySQL Server.

Motivo de elección

Es compatible con Django, además de tener conocimientos con su manejo

**Costos y Presupuestos**

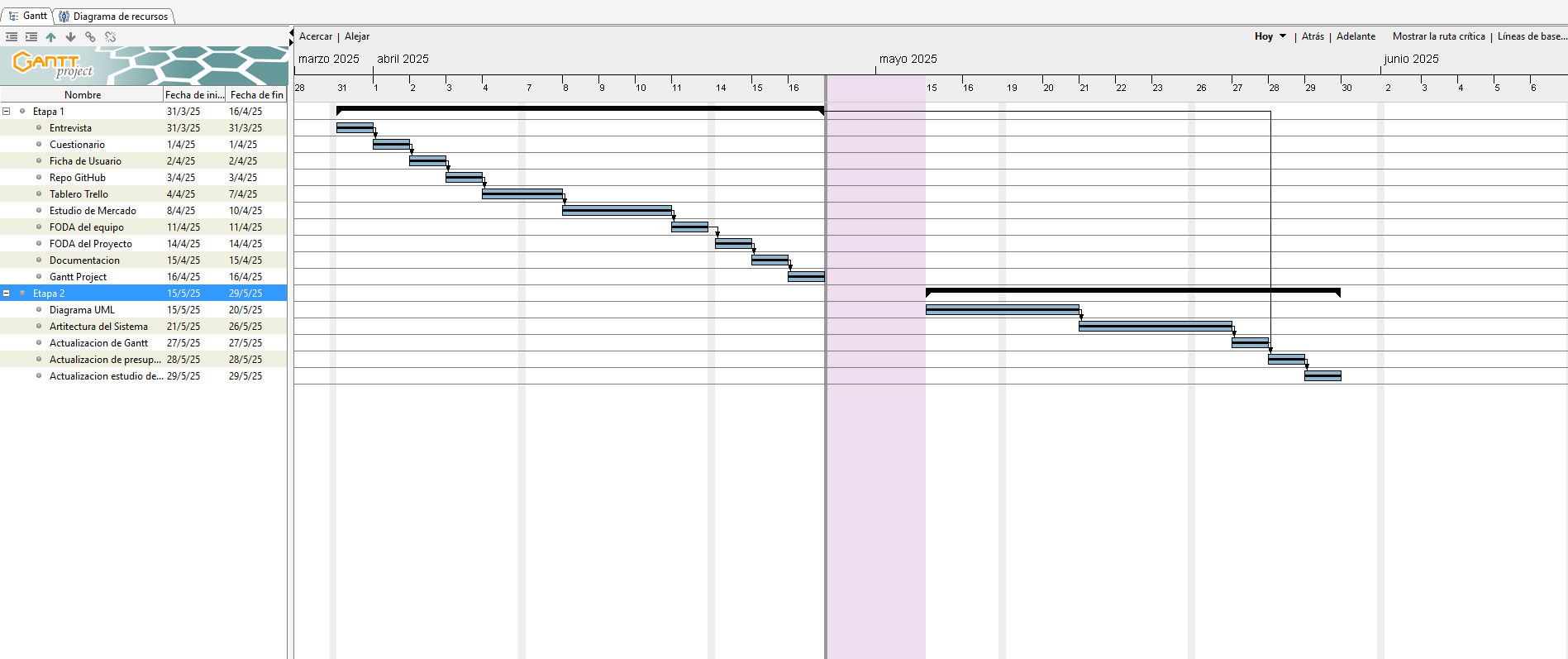
Todas las herramientas decididas son de uso gratuito, y no generan costos, con licencias y servicios de pago, pero en el presupuesto general se mira el tiempo y los recursos a:

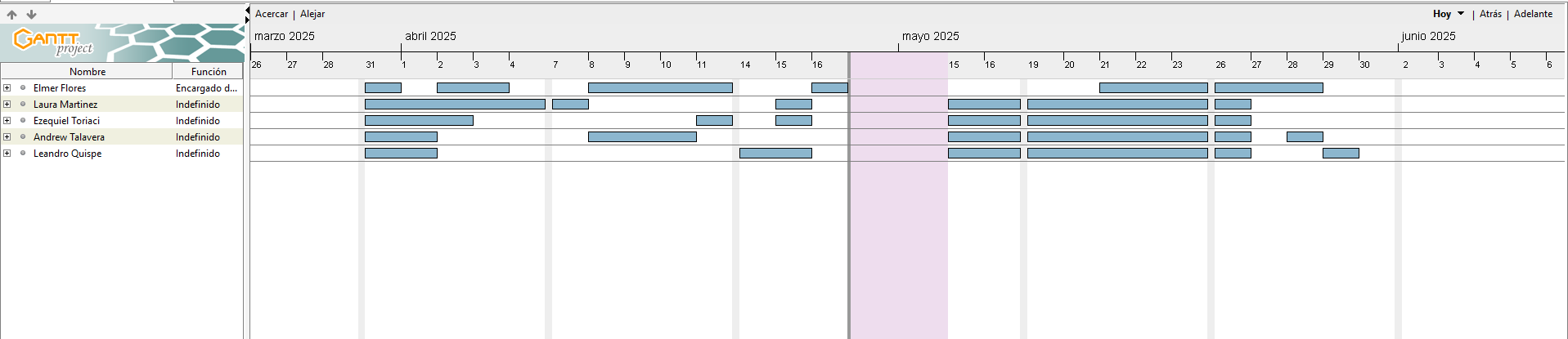
* **La instalacion y configuracion de los entornos de desarrollos( Visual studio Code, Python, Django, MySQL):** una 1:30 hs
* **La instrucción básica en caso de que un miembro del equipo no cuente con la experiencia necesaria de un entorno:** 1 o 3 días

**Planificación**

#### 

#### **Segundo sprint**





#### **Tercer sprint**

