

UNIDAD 6

Elaboración de diagramas de comportamiento

Contenidos

- 6.1. Diagramas de comportamiento
- 6.2. Diagramas de casos de uso
- 6.3. Diagramas de interacción
- 6.4. Diagramas de estados
- 6.5. Diagramas de actividades

6.1. Diagramas de comportamiento

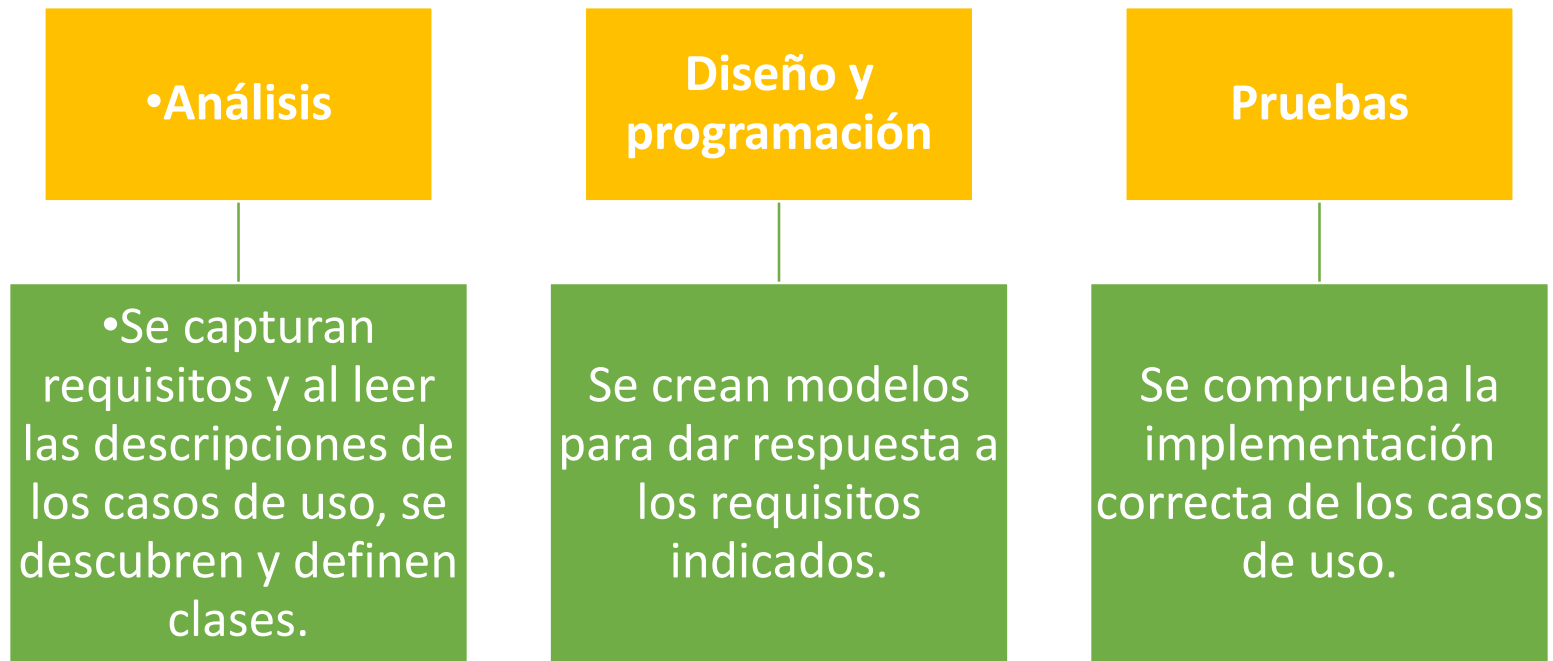
Sirven para visualizar, especificar, construir y documentar los aspectos dinámicos de un sistema

Tabla 6.1. Tipos de diagramas de comportamiento que se pueden construir en lenguaje UML y su uso

Diagrama	Uso
Diagramas de casos de uso	Describen las funcionalidades del sistema desde el punto de vista de la persona usuaria. Son fundamentales, pues se usan como punto de partida para los demás diagramas.
Diagramas de actividades	Se emplean para especificar paso a paso una operación compleja. Sirven para modelar el flujo de un caso de uso o entre casos de uso.
Diagramas de estados	Sirven para modelar el comportamiento de un objeto dirigido por eventos que provocan cambios de estado o transiciones.
Diagramas de secuencia	Muestran la secuencia cronológica de mensajes entre objetos durante un escenario de un caso de uso.
Diagramas de colaboración	Muestran un conjunto de objetos, los enlaces entre ellos y los mensajes que se reciben y se envían entre ellos.
Diagramas de tiempos	Representan el comportamiento de distintos objetos en un determinado periodo de tiempo.
Diagrama global de interacciones	Aportan una visión general de la interacción en el sistema.

6.2. Diagramas de casos de uso

- Utilidad de los diagramas de casos de uso:



- Proporcionan trazabilidad al sistema.

❑ Elementos de los diagramas de casos de uso:

- Casos de uso: descripciones de las funciones del sistema.
- Actores: usuarios del sistema, personas responsables de su uso o mantenimiento y otros sistemas con los que se interactúa.

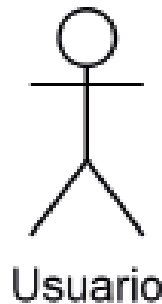


Figura 6.1. Símbolo que representa a un actor en un diagrama de casos de uso con su nombre debajo.

Relaciones:

- Comunicación entre un actor y un caso de uso.
- Uso o inclusión: un caso de uso incluye acciones de otro caso de uso.
- Extensión: un caso de uso incluye pasos adicionales respecto a otro.

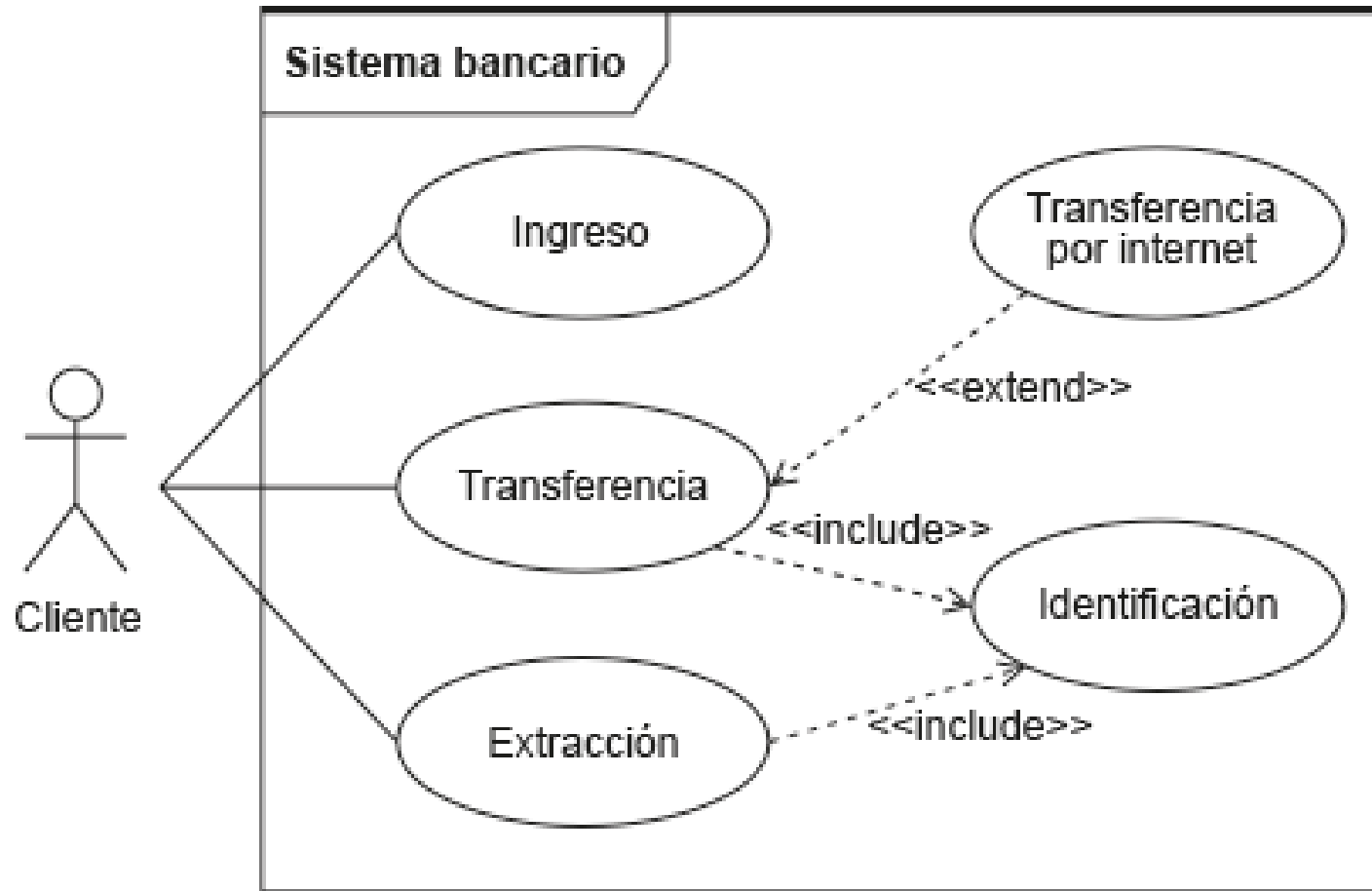
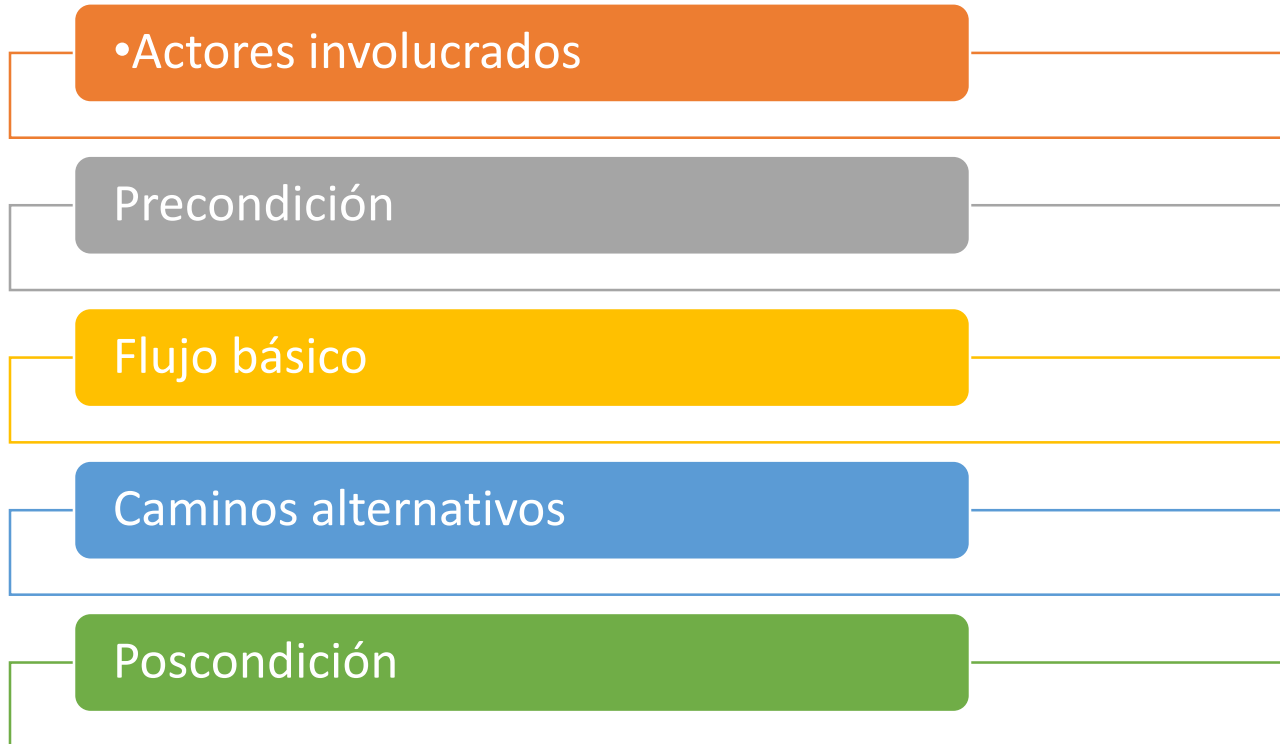


Figura 6.2. Diagrama de casos de uso para un sistema bancario en el que las y los clientes pueden realizar ingresos, extracciones o transferencias de dinero.

❑ Descripción de un caso de uso:



❑ Los casos de uso permiten realizar una verificación y validación del sistema.

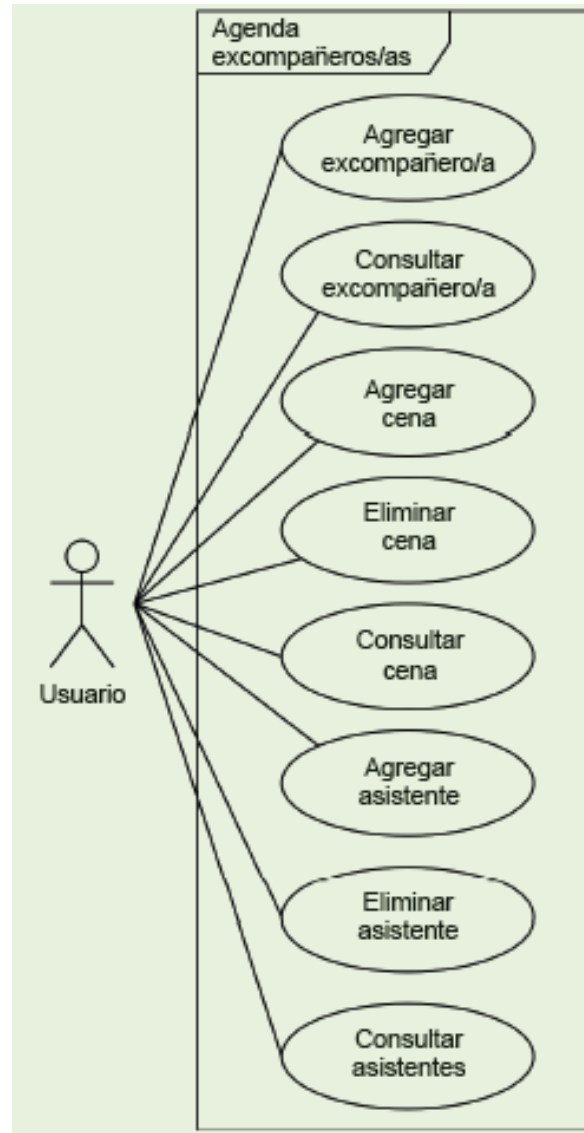


Figura 6.3. Diagrama de casos de uso para un sistema en el que se desean almacenar datos de antiguos compañeros y compañeras con los que se celebran cenas anualmente.

Descripción del caso de uso Agregar excompañero/a

- *Actores involucrados:* persona usuaria.
- *Precondición:* la agenda tiene que estar creada.
- *Flujo básico:*
 1. La persona usuaria solicita agregar un nuevo excompañero o excompañera.
 2. Se pide a la persona usuaria de la aplicación la introducción del nombre completo, teléfono y correo electrónico del nuevo excompañero o excompañera.
 3. Tras la introducción de los datos, el sistema comprueba que no haya una persona con el mismo nombre y, si es así, lo añade a la agenda y muestra un mensaje de confirmación en pantalla.
 4. Se vuelve a solicitar al usuario o usuaria la selección de una operación.
- *Caminos alternativos:*
 - Si en el paso 3 se detecta que ya existe un excompañero o excompañera con ese nombre, se muestra un mensaje que el usuario debe aceptar y se pasa al paso 4 anterior.
- *Poscondición:* la agenda dispone de un excompañero o excompañera más o del mismo de número de antiguas y antiguos compañeros si la persona ya estaba en la agenda.

Descripción del caso de uso Agregar cena

■ *Actores involucrados:* persona usuaria.

■ *Precondición:* la agenda tiene que estar creada y debe existir algún excompañero agregado o excompañera agregada.

■ *Flujo básico:*

1. El usuario o la usuaria solicita introducir una nueva cena.
2. Se pide a la persona usuaria la introducción de la fecha y el lugar de la cena.
3. El sistema comprueba que no haya ninguna cena ese mismo año.
4. Se pide a la persona usuaria la introducción del nombre de la persona organizadora de la cena.
5. El sistema comprueba que haya alguna persona con el nombre del organizador de la cena y si la hay, se añaden los datos de la cena y se muestra un mensaje de confirmación en la pantalla.
6. Se vuelve a solicitar al usuario o usuaria que seleccione una operación.

■ *Caminos alternativos:*

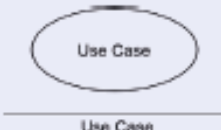

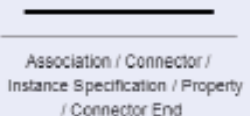
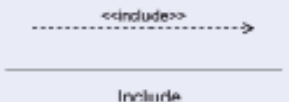

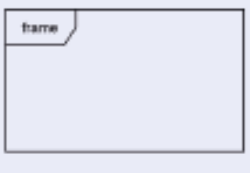
- Si en el paso 3 se detecta que ya existe una cena ese año, se muestra un mensaje que el usuario o usuaria debe aceptar y se pasa al paso 6.
- Si en el paso 5 se detecta que no existe ninguna persona con el nombre de la persona organizadora, se muestra un mensaje que la persona usuaria debe aceptar y se pasa al paso 6.

■ *Poscondición:* la agenda dispone de una cena más o del mismo de número de cenas si esta no se ha podido añadir.

6.2.1. Herramientas para la elaboración de diagramas de casos de uso

**Elaboración
de diagramas de
casos de uso con
diagrams.net**

Tabla 6.2. Elementos que forman parte de los diagramas de casos de uso con su icono y paleta correspondiente en diagrams.net

Elemento	Icono	Paleta
Caso de uso		UML
Actor		UML
Relación de comunicación		UML 2.5
Relación de inclusión		UML 2.5
Relación de extensión		UML 2.5
Sistema		UML

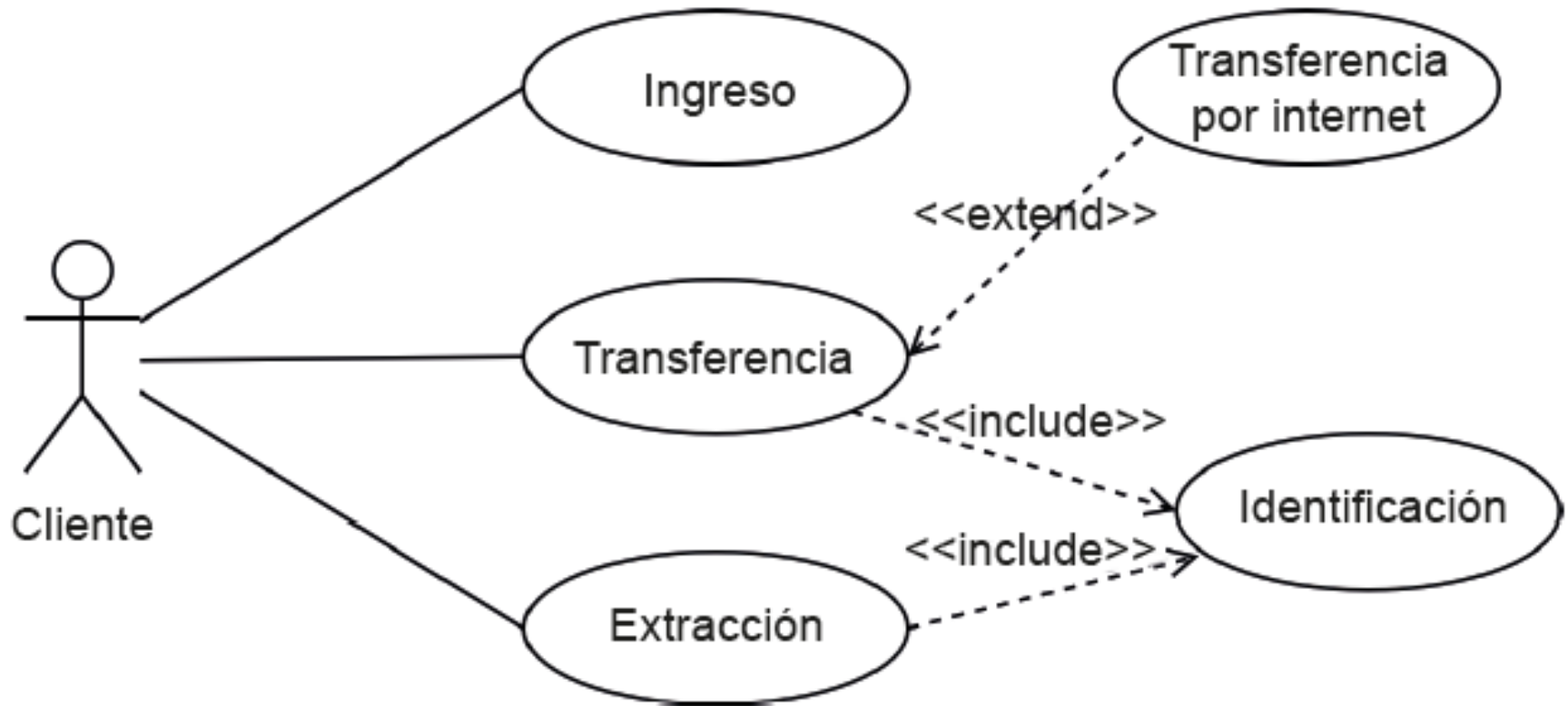








Figura 6.4. Diagrama de casos de uso para una entidad bancaria creado con diagrams.net.

Elaboración de diagramas de casos de uso con Papyrus

Tabla 6.3. Elementos que forman parte de los diagramas de casos de uso con el símbolo correspondiente en Papyrus SysML y la sección de la paleta donde se encuentra

Elemento	Símbolo	Sección de la paleta
Caso de uso	 Use Case	<i>Nodes</i>
Actor	 Actor	<i>Nodes</i>
Relación de comunicación	 Association	<i>Links</i>
Relación de inclusión	 Include	<i>Links</i>
Relación de extensión	 Extend	<i>Links</i>
Sistema	 Subject	<i>Nodes</i>

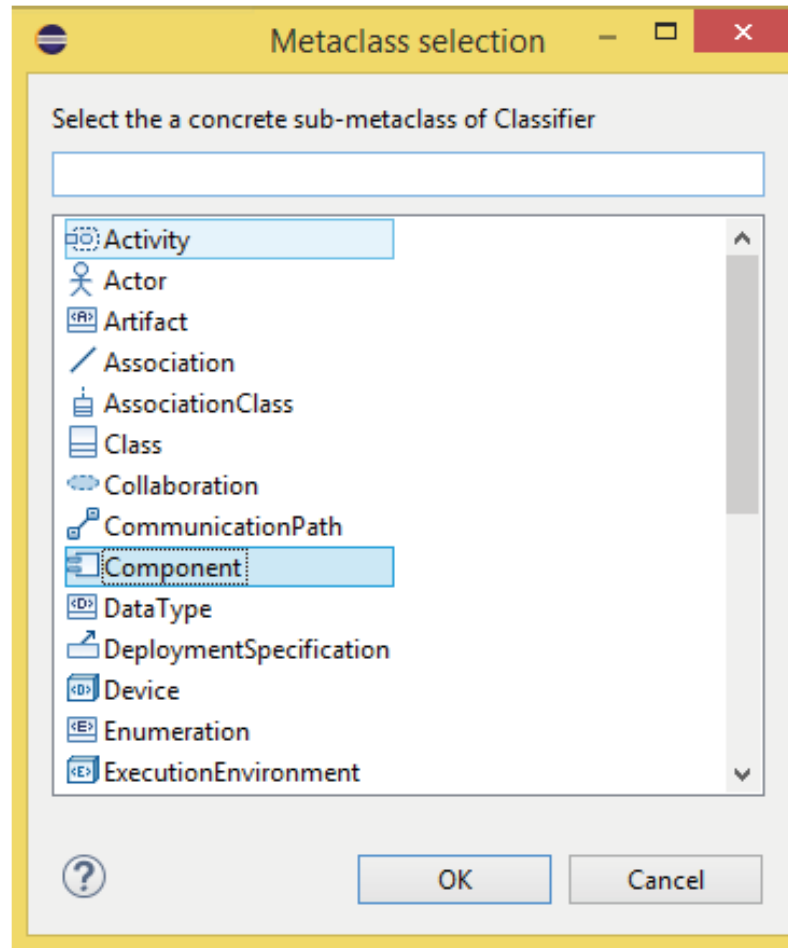


Figura 6.5. Ventana en la que se indica el tipo de elemento que se desea crear cuando se establecen los límites del sistema con el elemento *Subject* de la paleta. Se debe elegir el elemento *Component*.

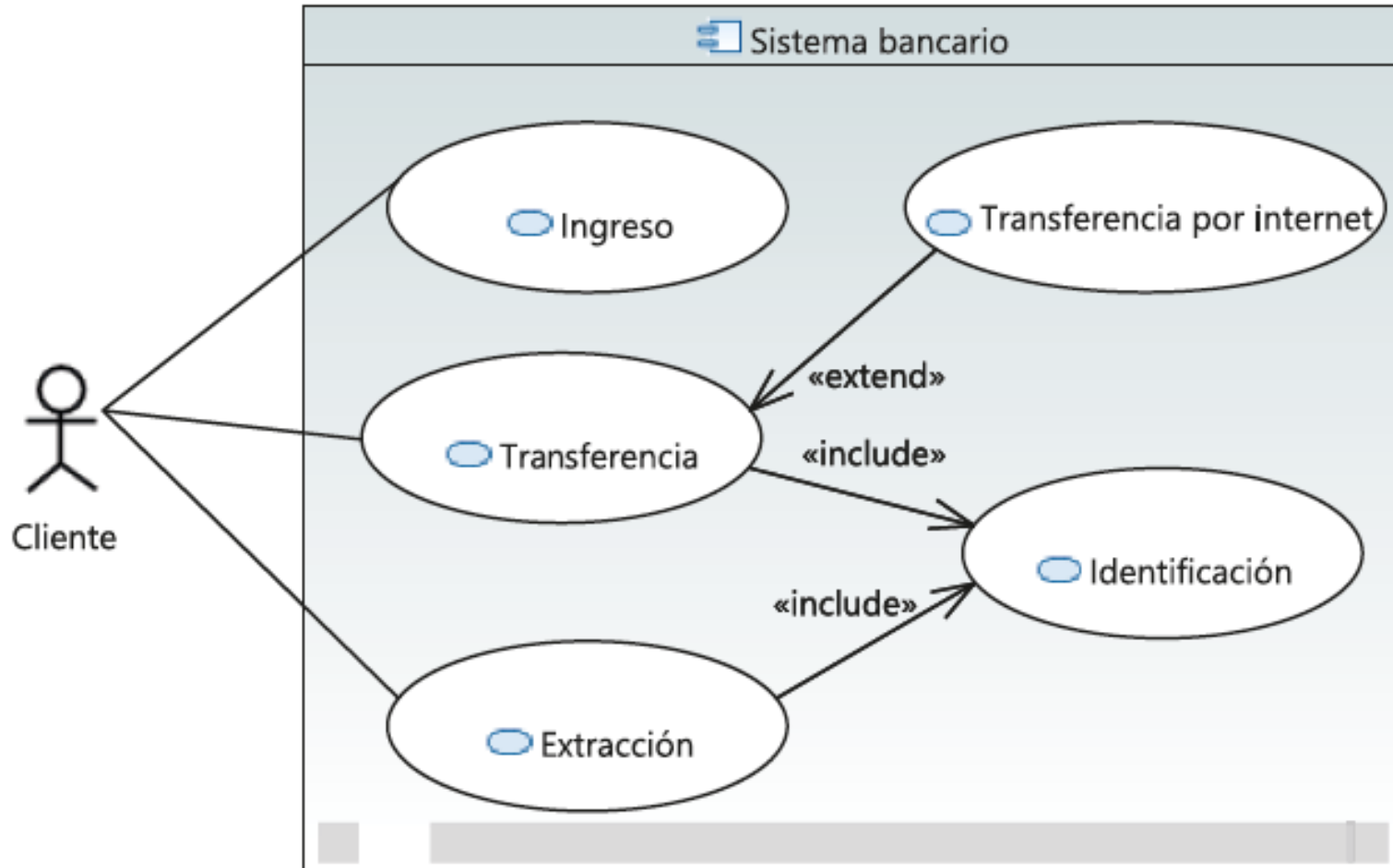


Figura 6.6. Diagrama de casos de uso para una entidad bancaria creado con Papyrus SysML en Eclipse.

6.3. Diagramas de interacción

Muestran un conjunto de objetos y los mensajes que se intercambian.

6.3.1. Diagramas de secuencia

- ❑ Muestran la secuencia cronológica de mensajes que se intercambian los objetos durante un escenario de un caso de uso.
- ❑ Elementos:
 - **Línea de vida** (línea con trazado discontinuo). En la cima se dibuja un rectángulo para una clase o un actor.
 - **Mensajes (flechas)**: solicitud de ejecución de un método a la clase receptora.
 - **Tiempo durante el cual está activo un método** (rectángulo sobre la línea de vida).

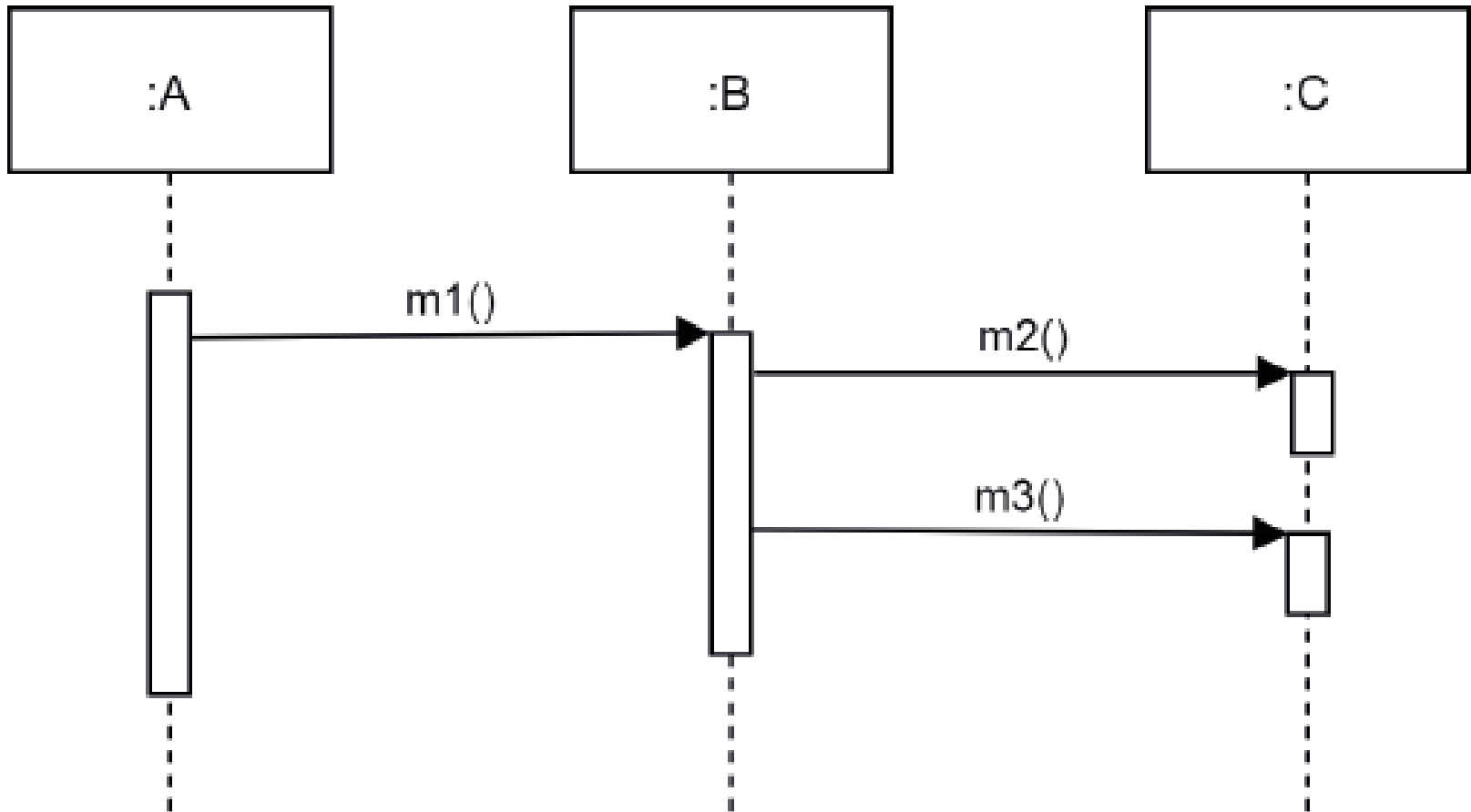


Figura 6.7. Diagrama de secuencia de ejemplo con tres objetos que se intercambian mensajes para realizar lo especificado en un supuesto escenario de un caso de uso.

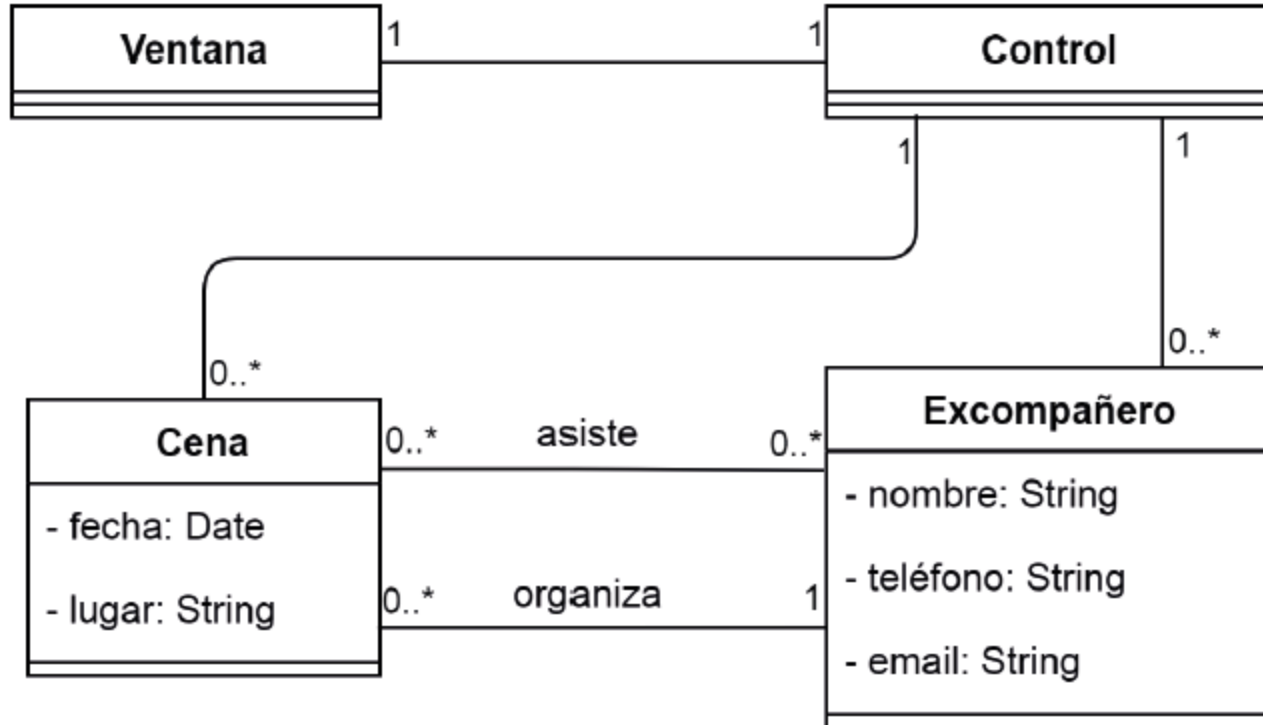


Figura 6.8. Diagrama de clases de análisis con las clases de interfaz, de control y de entidad identificadas en la aplicación para una agenda de antiguos compañeros y compañeras de estudios.

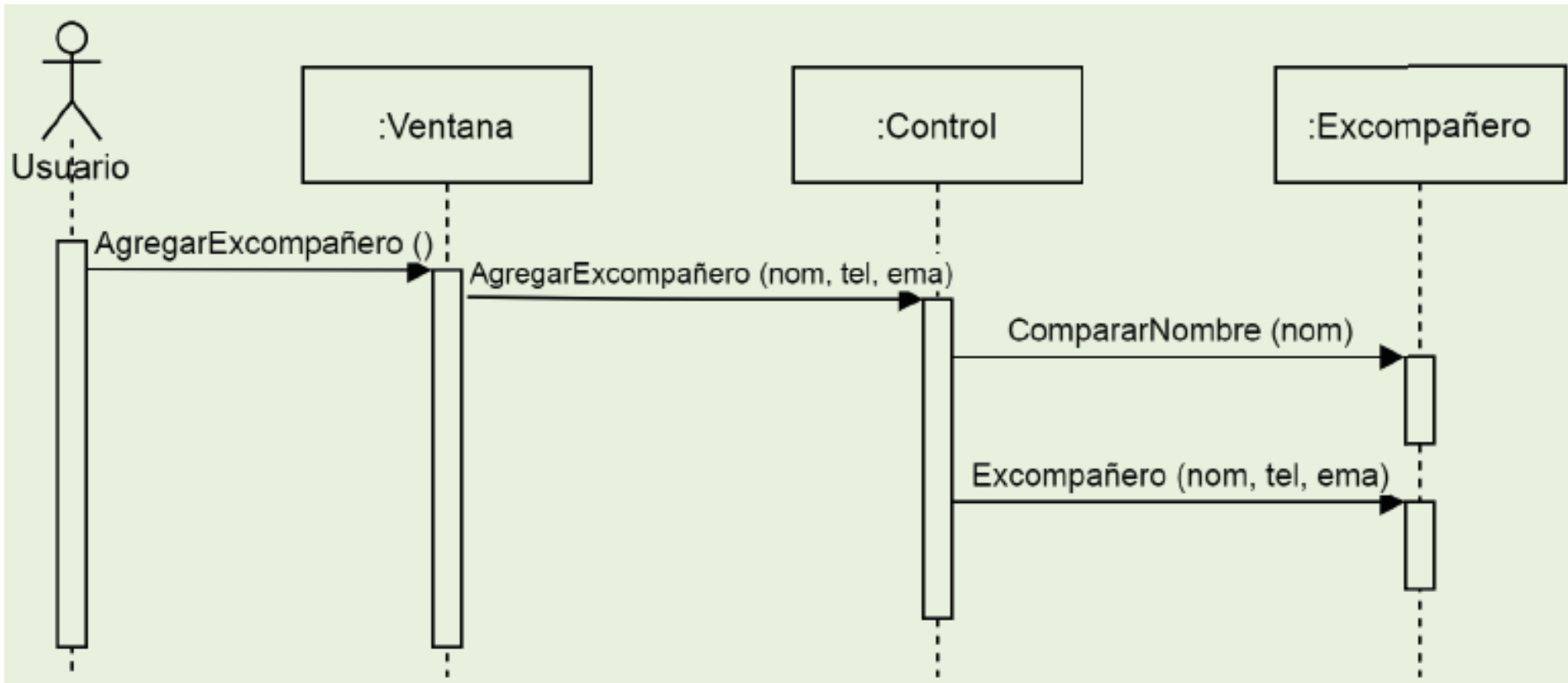


Figura 6.9. Diagrama de secuencia del caso de uso *Agregar excompañero/a*.

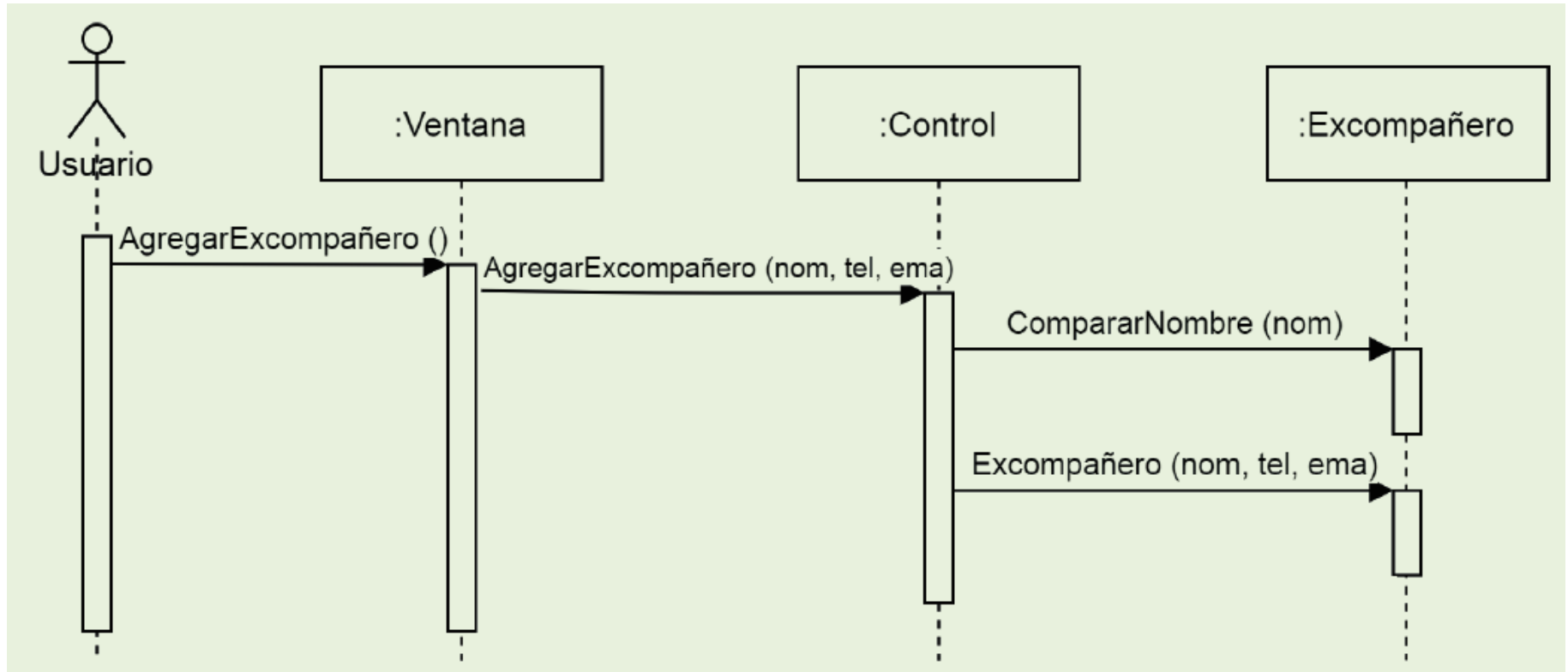


Figura 6.10. Diagrama de secuencia del caso de uso *Consultar excompañero/a*.

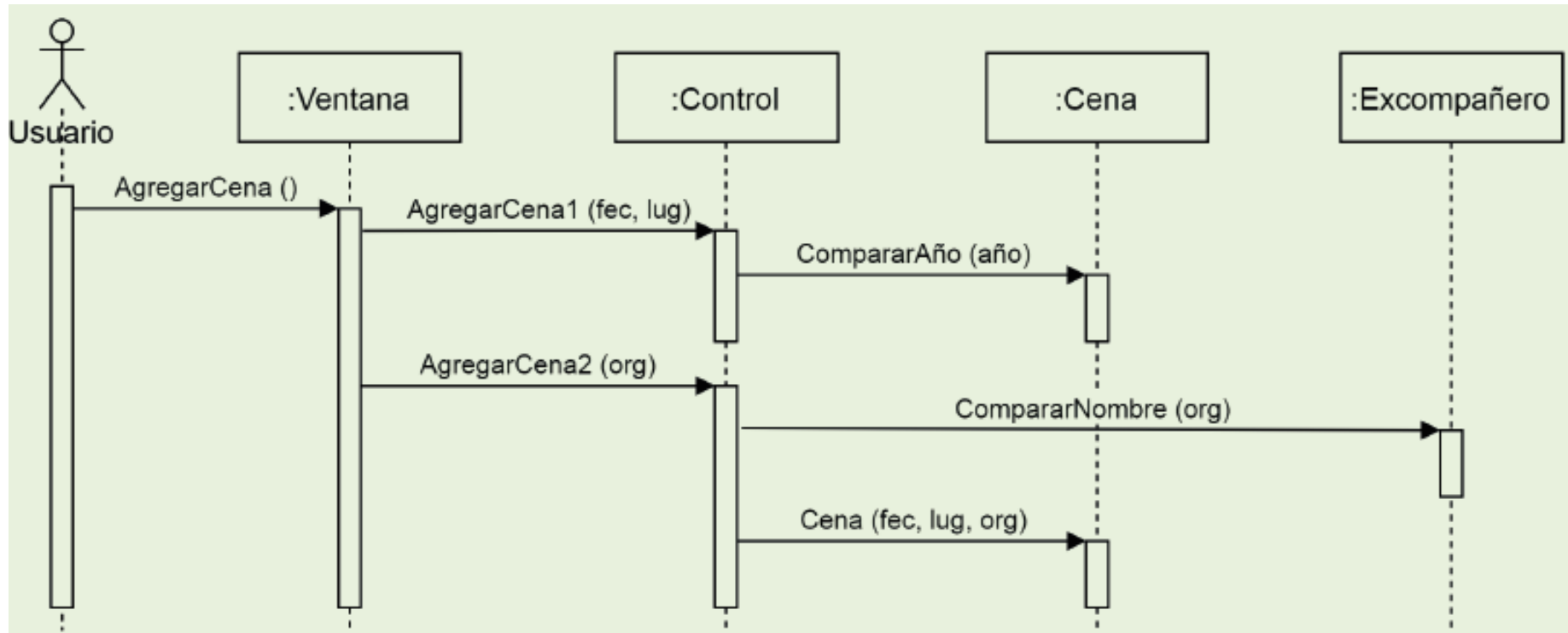


Figura 6.11. Diagrama de secuencia del caso de uso *Agregar cena*.

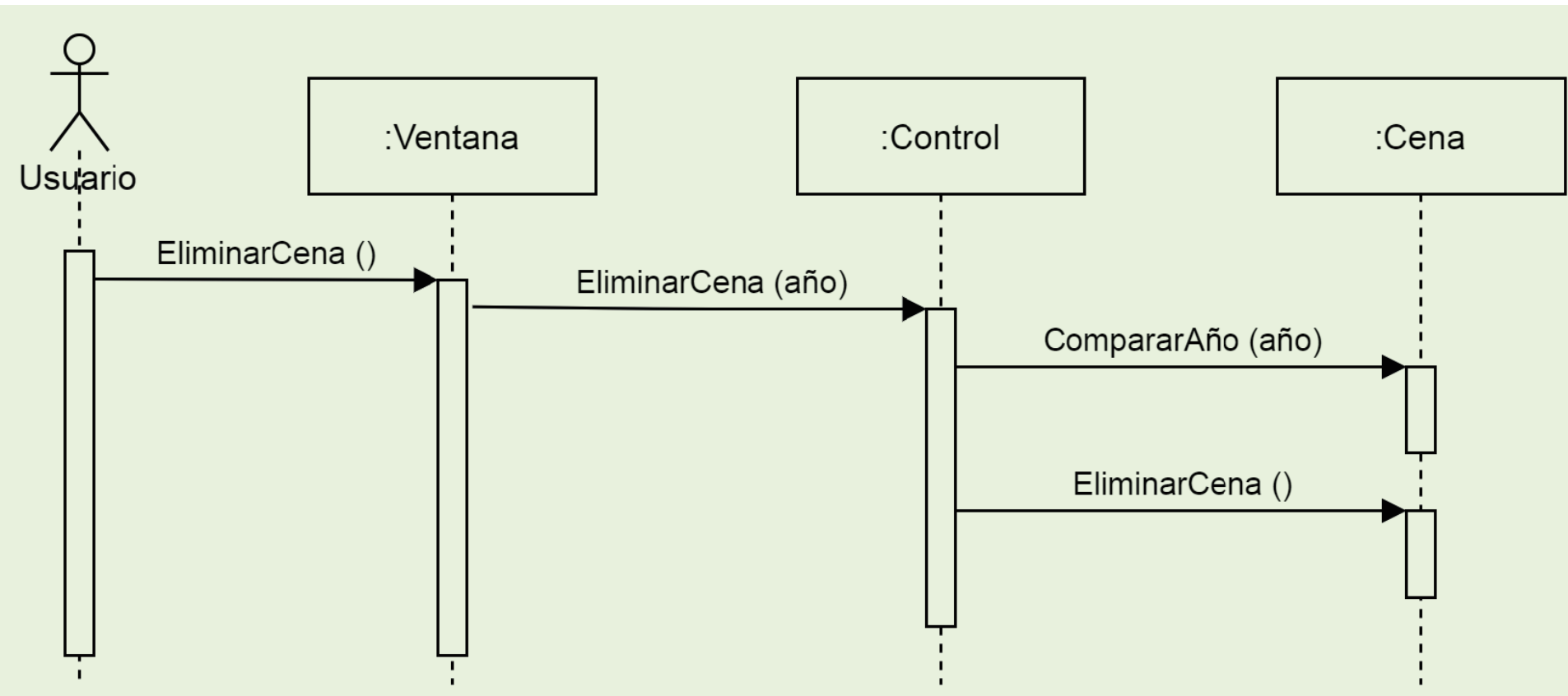


Figura 6.12. Diagrama de secuencia del caso de uso *Eliminar cena*.

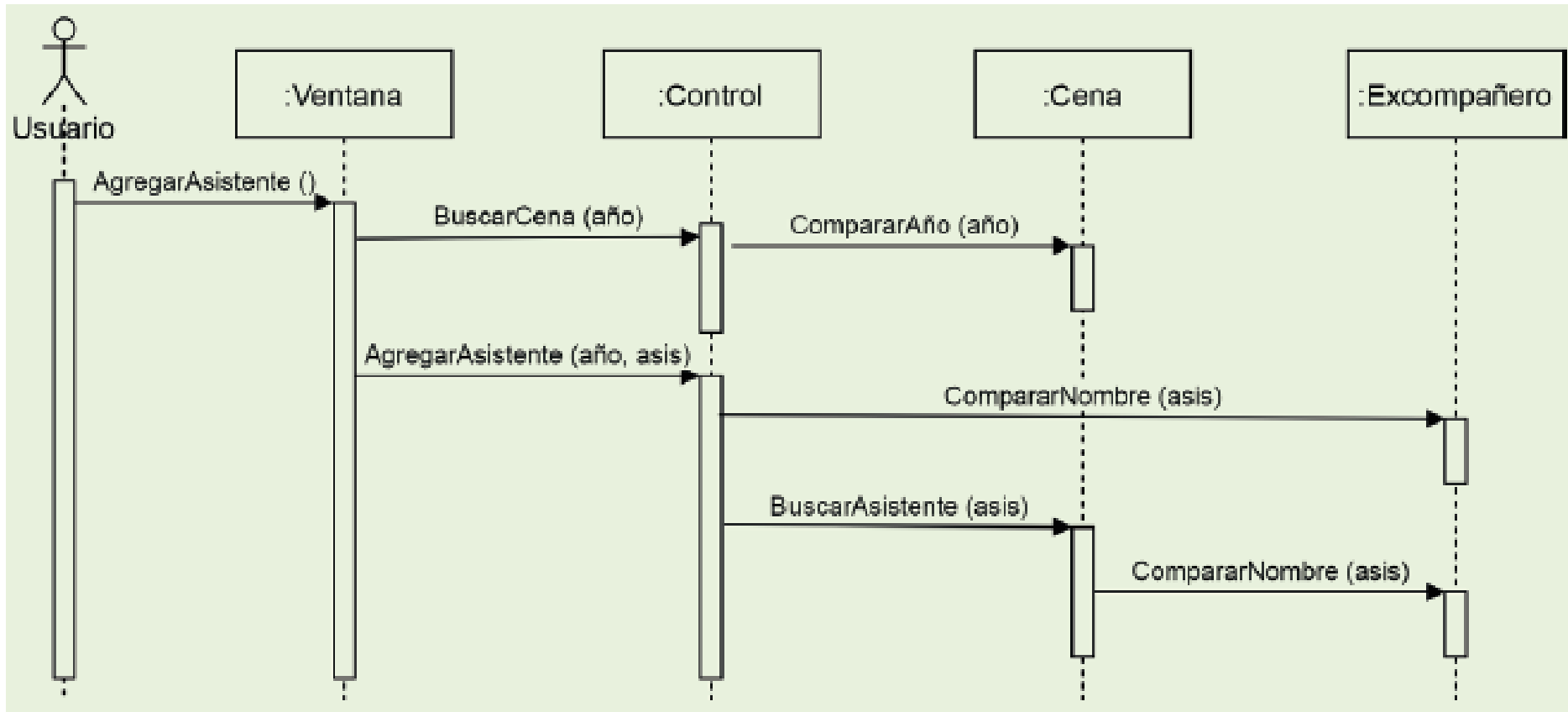


Figura 6.13. Diagrama de secuencia correspondiente al caso de uso *Agregar asistente*.

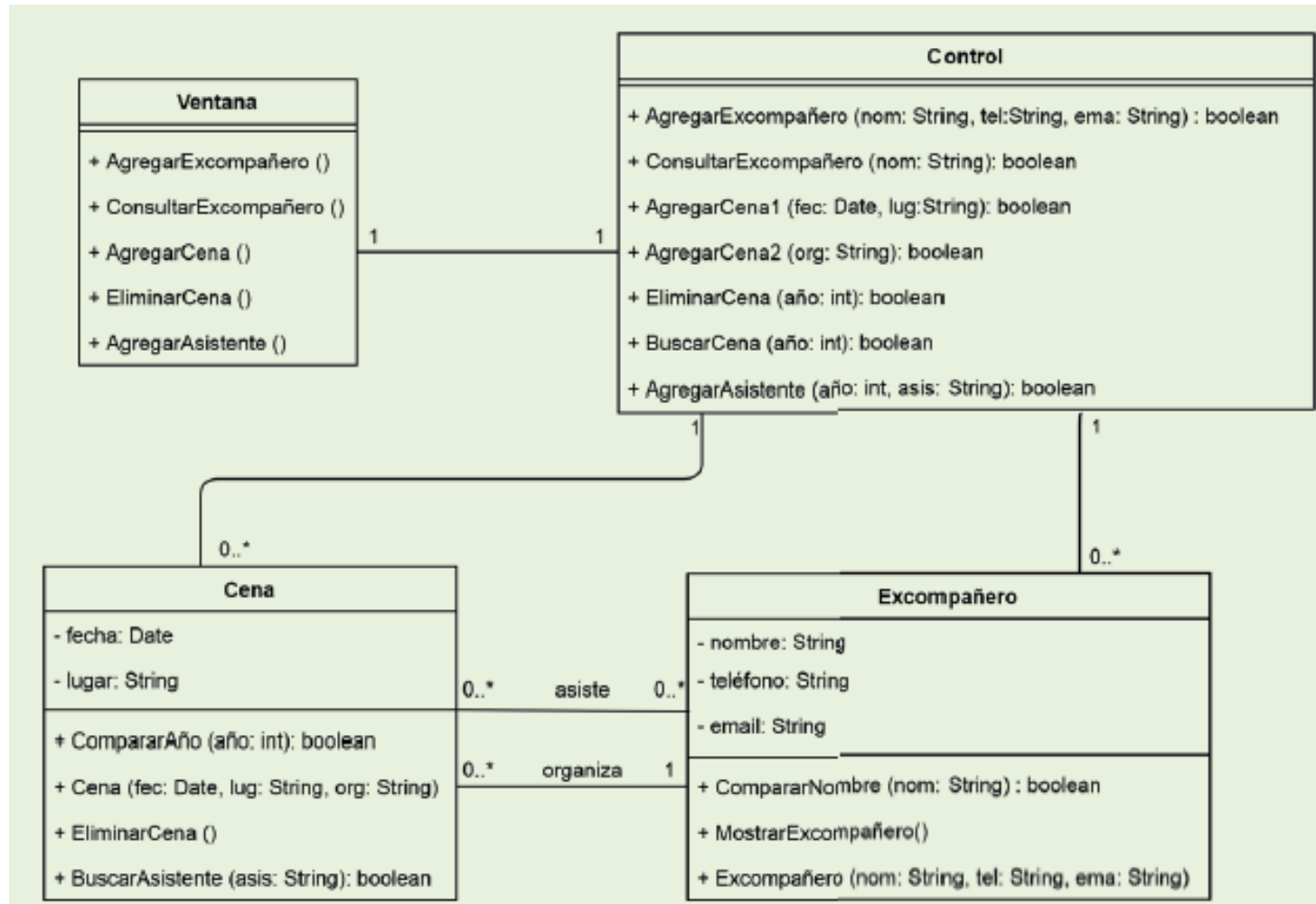


Figura 6.14. Diagrama de clases ampliado con los métodos encontrados en los diagramas de secuencia correspondientes a los casos de uso *Agregar excompañero/a*, *Consultar excompañero/a*, *Agregar cena*, *Eliminar cena* y *Agregar asistente*

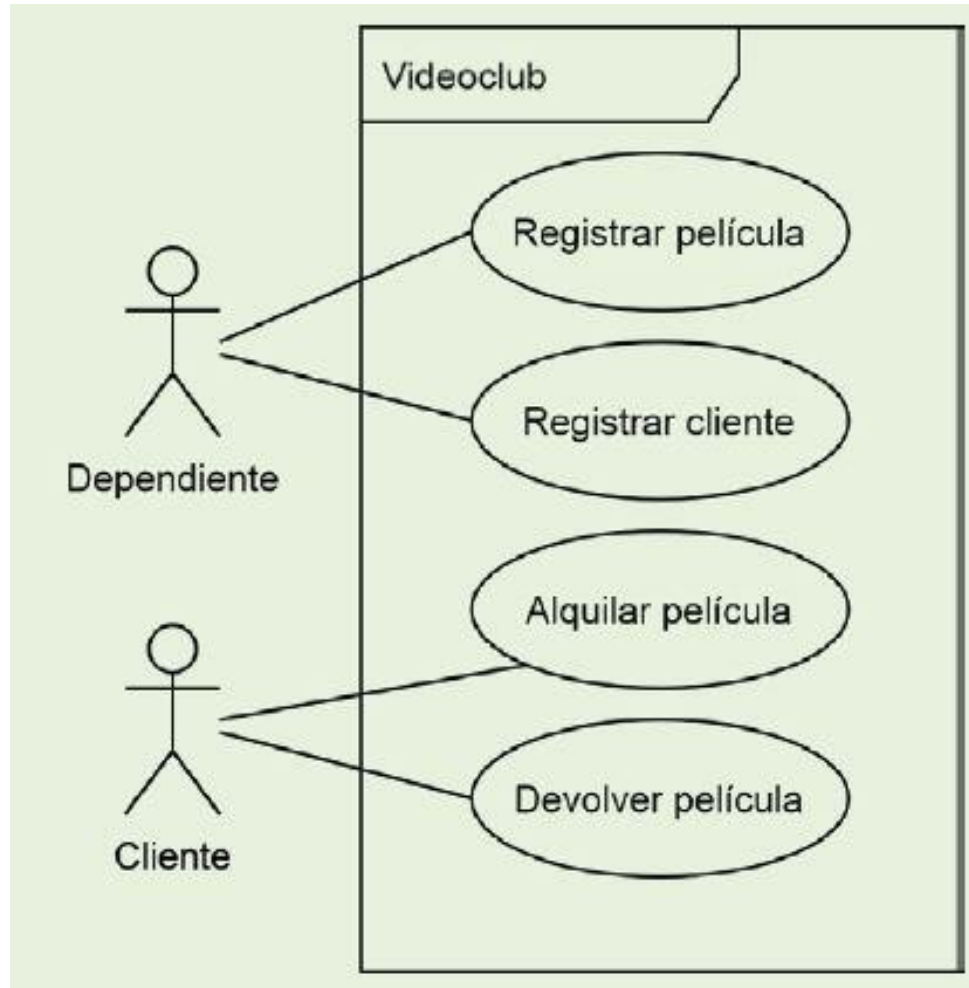


Figura 6.15. Diagrama de casos de uso para un videoclub, que incluye los dos actores que intervienen en la aplicación (cliente y dependiente) y cuatro casos de uso (registrar película, registrar cliente, alquilar película y devolver película).

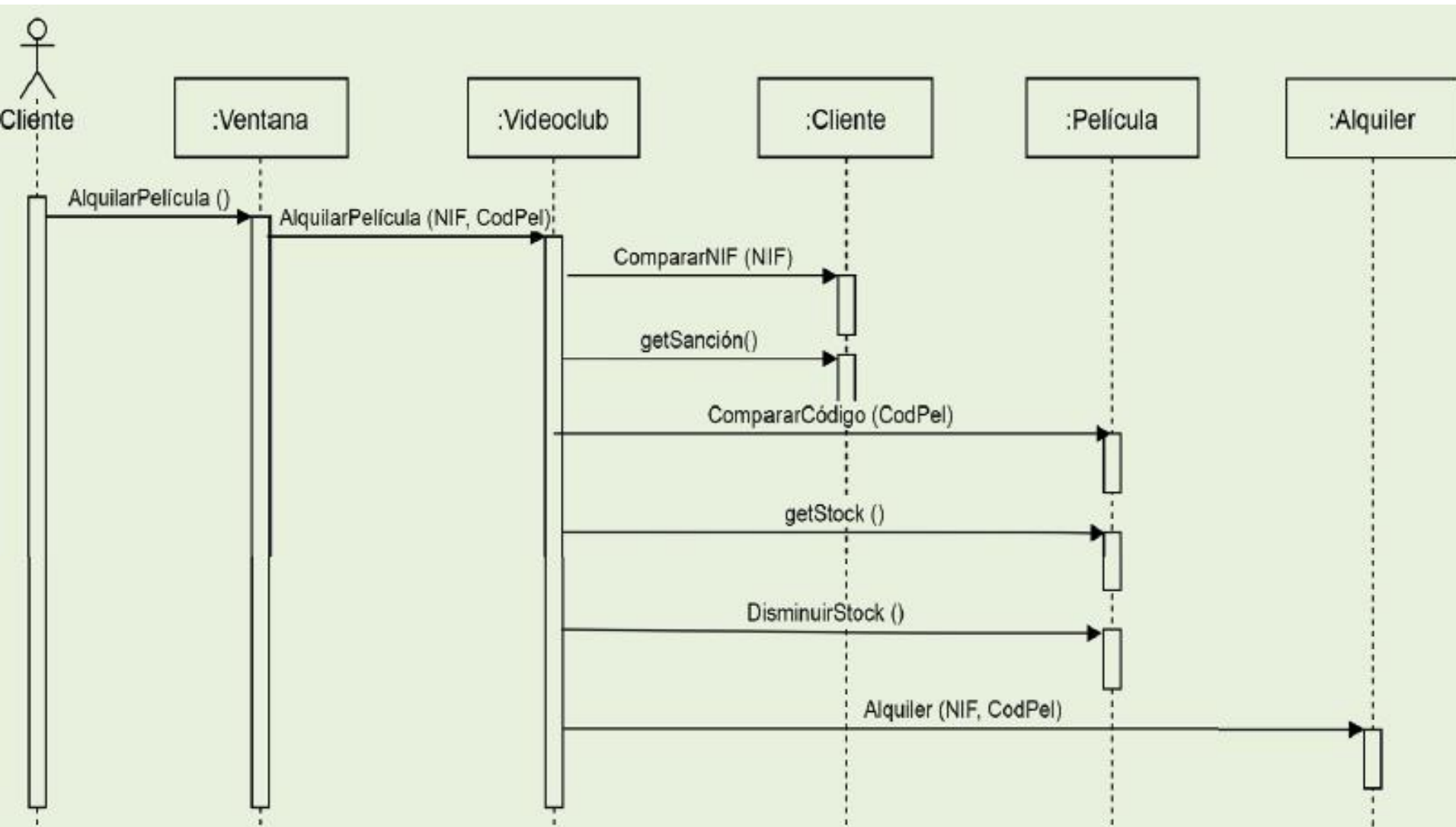


Figura 6.16. Diagrama de secuencia correspondiente al caso de uso *Alquilar película*.

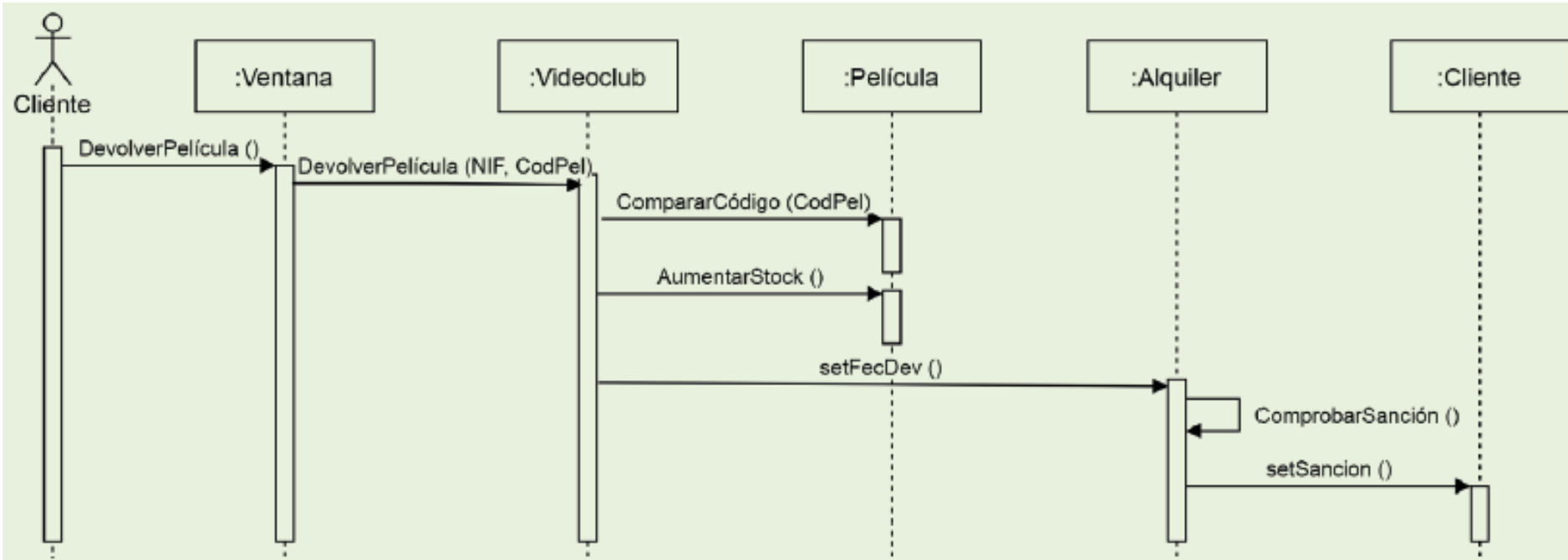


Figura 6.17. Diagrama de secuencia del caso de uso *Devolver película*.

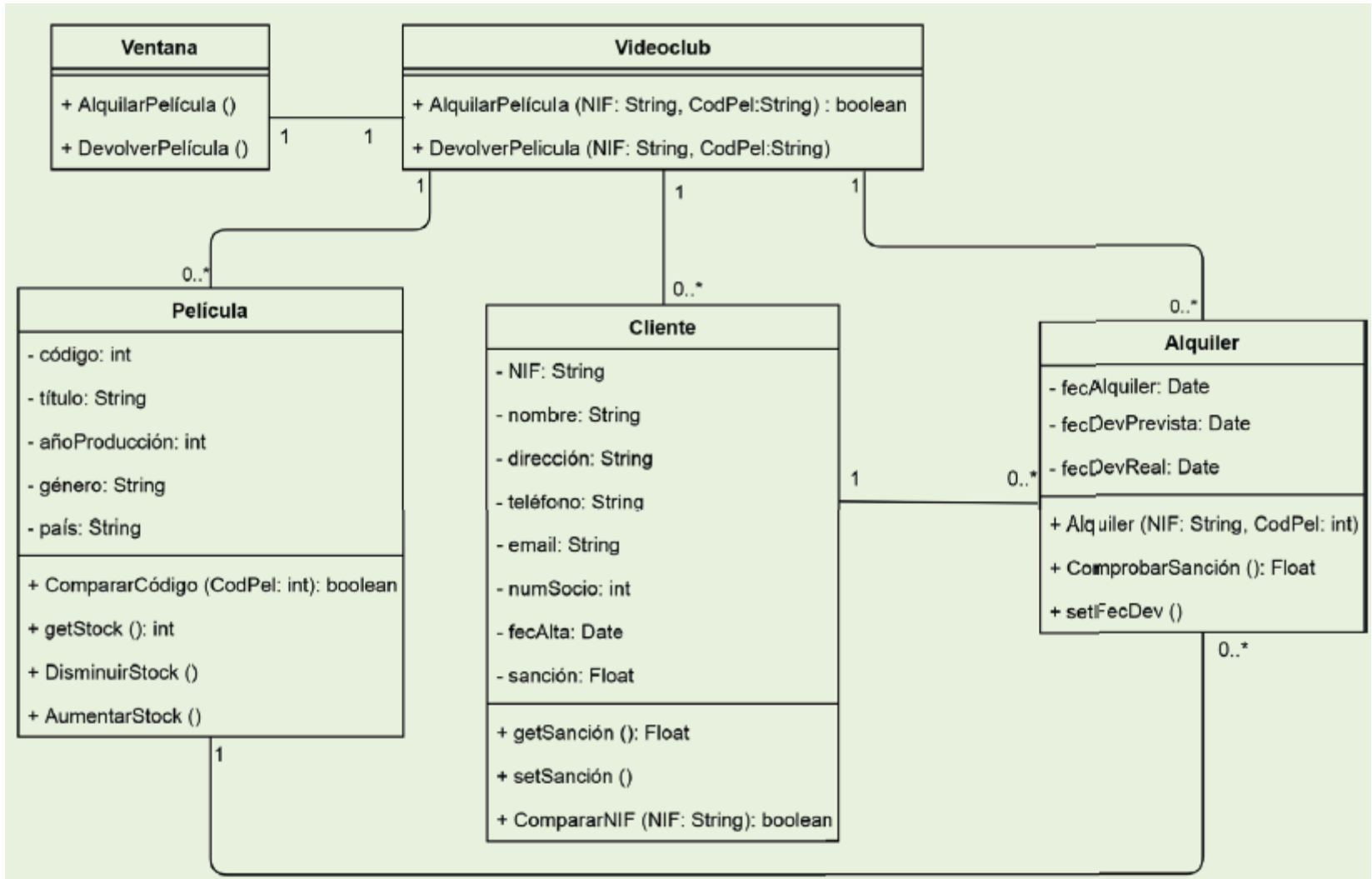


Figura 6.18. Diagrama de clases para un videoclub.

6.3.2. Diagramas de colaboración

- ❑ Muestran varias clases que colaboran y los mensajes que se intercambian.
- ❑ Elementos:

- Objetos, clases o actores.
- Enlaces (líneas entre clases).
- Mensajes (flechas): delante de cada mensaje se escribe un número que indica el orden.

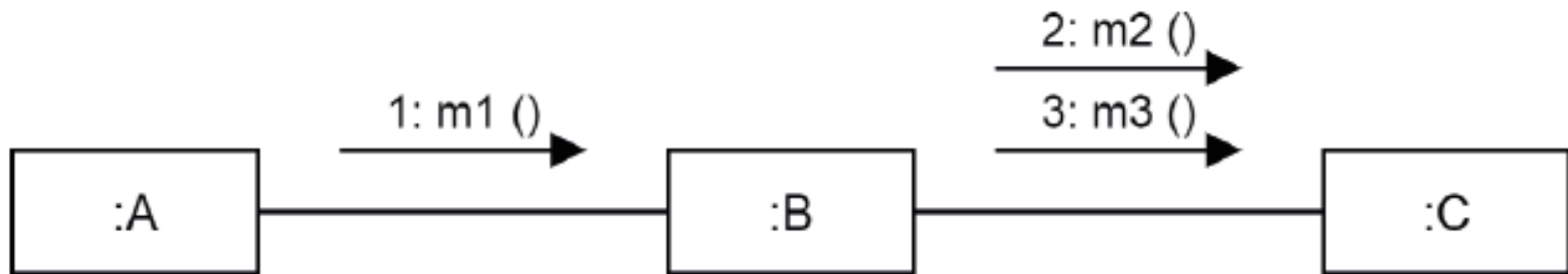


Figura 6.19. Diagrama de colaboración de ejemplo correspondiente al diagrama de secuencia de la Figura 6.7.

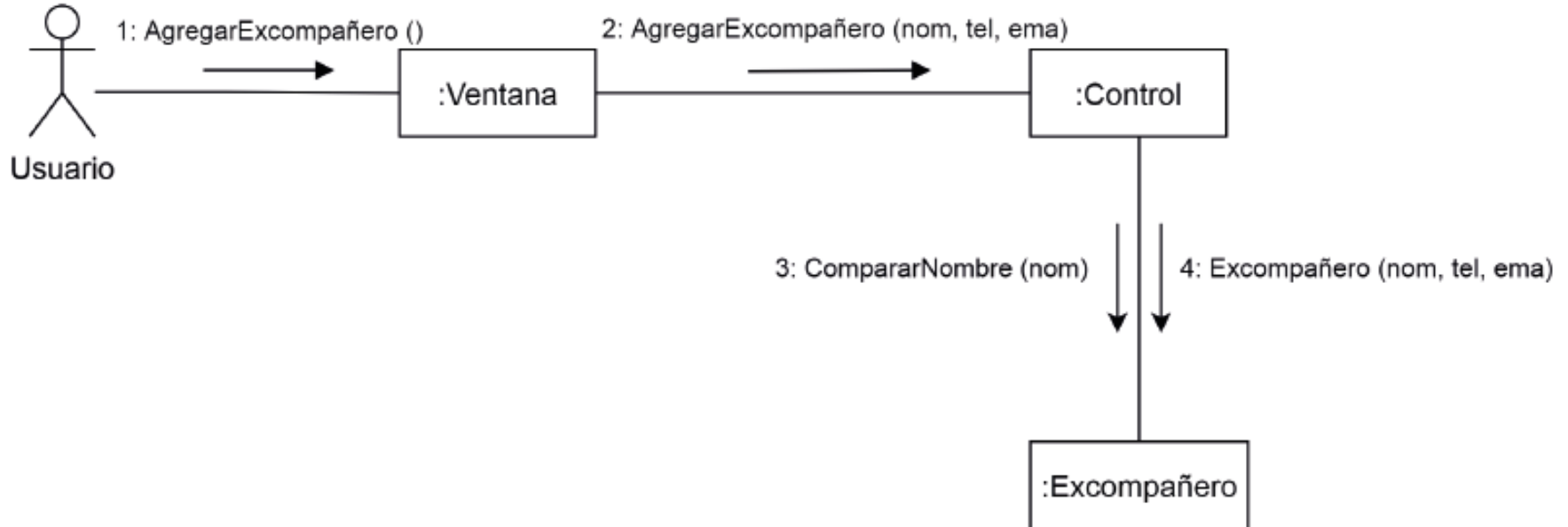


Figura 6.20. Diagrama de colaboración de ejemplo correspondiente al diagrama de secuencia de la Figura 6.9.

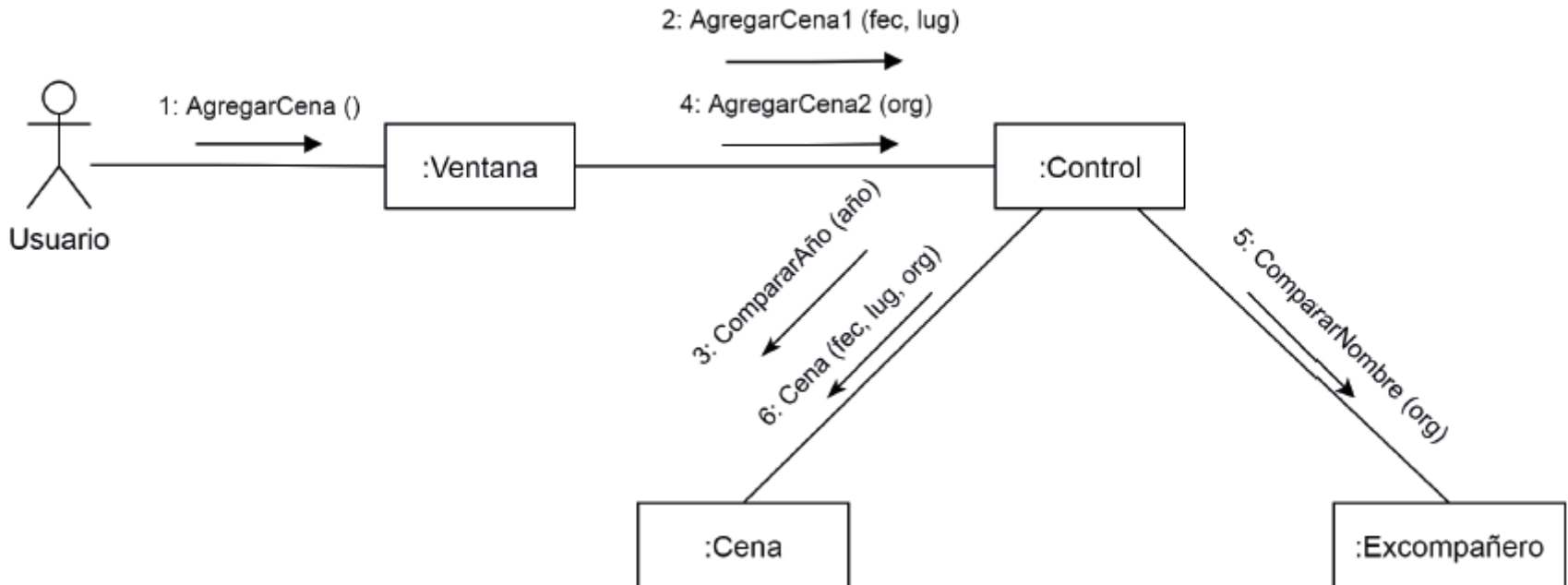

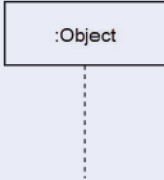
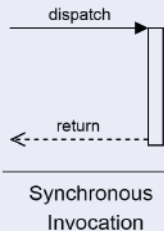
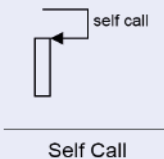


Figura 6.21. Diagrama de colaboración de ejemplo correspondiente al diagrama de secuencia de la Figura 6.11.

6.3.3. Herramientas para la elaboración de diagramas de interacción

Elaboración de diagramas de secuencia con diagrams.net

Tabla 6.4. Elementos que forman parte de los diagramas de secuencia, su icono correspondiente en diagrams.net y la paleta que le corresponde

Elemento	Icono	Paleta
Línea de vida para un actor		UML
Línea de vida para un objeto		UML
Mensaje enviado de un objeto a otro		UML
Mensaje enviado de un objeto a sí mismo		UML

Elaboración de diagramas de secuencia con Papyrus

1.º Se crea el siguiente diagrama de casos de uso en un nuevo proyecto

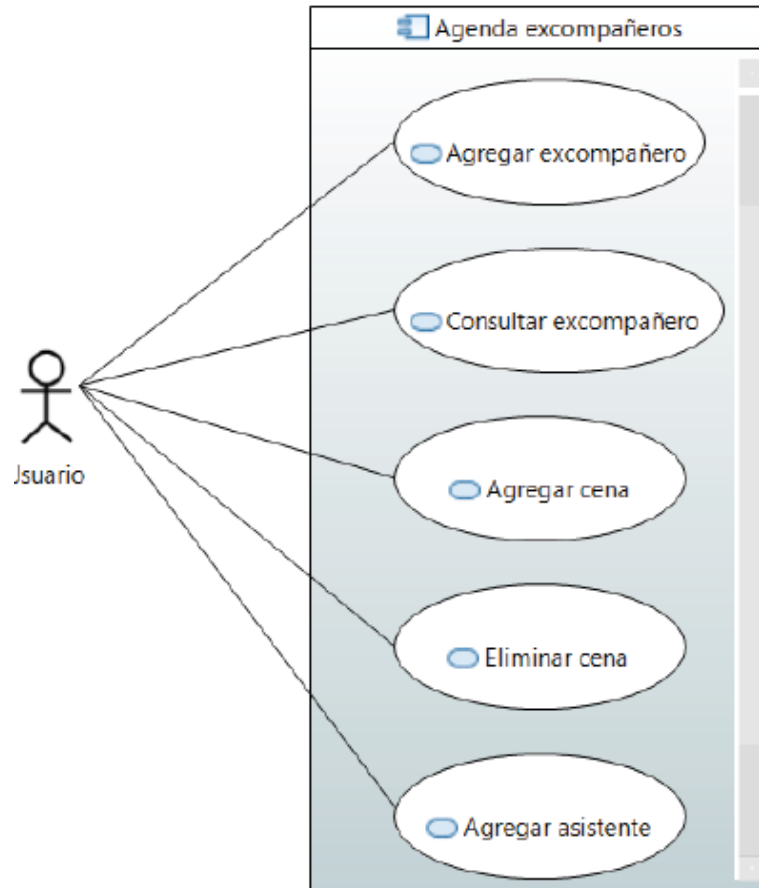


Figura 6.22. Diagrama de casos de uso para la aplicación de una agenda de antiguos compañeros y compañeras que incluye los casos de uso *Agregar excompañero/a*, *Consultar excompañero/a*, *Agregar cena*, *Eliminar cena* y *Eliminar asistente*.

2.º Se crea el diagrama de clases de la Figura 6.14, incluyendo métodos.

The image shows a 'Create a new Parameter' dialog box with the following fields and values:

- Name: nom
- Label: (empty)
- Is exception: ☐ true ☒ false
- Is stream: ☐ true ☒ false
- Direction: in (dropdown)
- Visibility: public (dropdown)
- Multiplicity: 1 (dropdown)
- Is ordered: ☐ true ☒ false
- Is unique: ☒ true ☐ false
- Effect: create (dropdown)
- Type: String (dropdown)
- Default value: <Undefined> (dropdown)

At the bottom, there are tabs for 'UML' and 'Comments', a help icon (?), and 'OK' and 'Cancel' buttons.

Figura 6.23. Propiedades del parámetro *nom* del método *AgregarExcompañero* de la clase *Control*. El nombre, dirección y tipo de dato son las más importantes.

The image shows a 'Create a new Parameter' dialog box. It has a yellow title bar with a small icon on the left and standard window controls on the right. The main area is light gray and contains the following fields and controls:

- Name:** A text input field.
- Label:** A text input field.
- Is exception:** Radio buttons for 'true' and 'false' (selected).
- Is stream:** Radio buttons for 'true' and 'false' (selected).
- Direction:** A dropdown menu with 'return' selected.
- Visibility:** A dropdown menu with 'public' selected.
- Multiplicity:** A dropdown menu with '1' selected.
- Is ordered:** Radio buttons for 'true' and 'false' (selected).
- Is unique:** Radio buttons for 'true' (selected) and 'false'.
- Effect:** A dropdown menu with 'create' selected.
- Type:** A dropdown menu with 'Boolean' selected. To its right are icons for creating, editing, and deleting types.
- Default value:** A text input field with '<Undefined>' selected. To its right are icons for creating, editing, and deleting default values.

At the bottom, there are two tabs: 'UML' (selected) and 'Comments'. Below the tabs is a help icon (a question mark in a circle). On the right side of the bottom are 'OK' and 'Cancel' buttons.

Figura 6.24. Ventana en la que se muestran las propiedades para el valor de retorno del método *AgregarExcompañero* de la clase *Control*. No es necesario asignarle nombre, pero se debe indicar en *Direction* el valor *return* y se debe indicar su tipo de dato, *Boolean*.

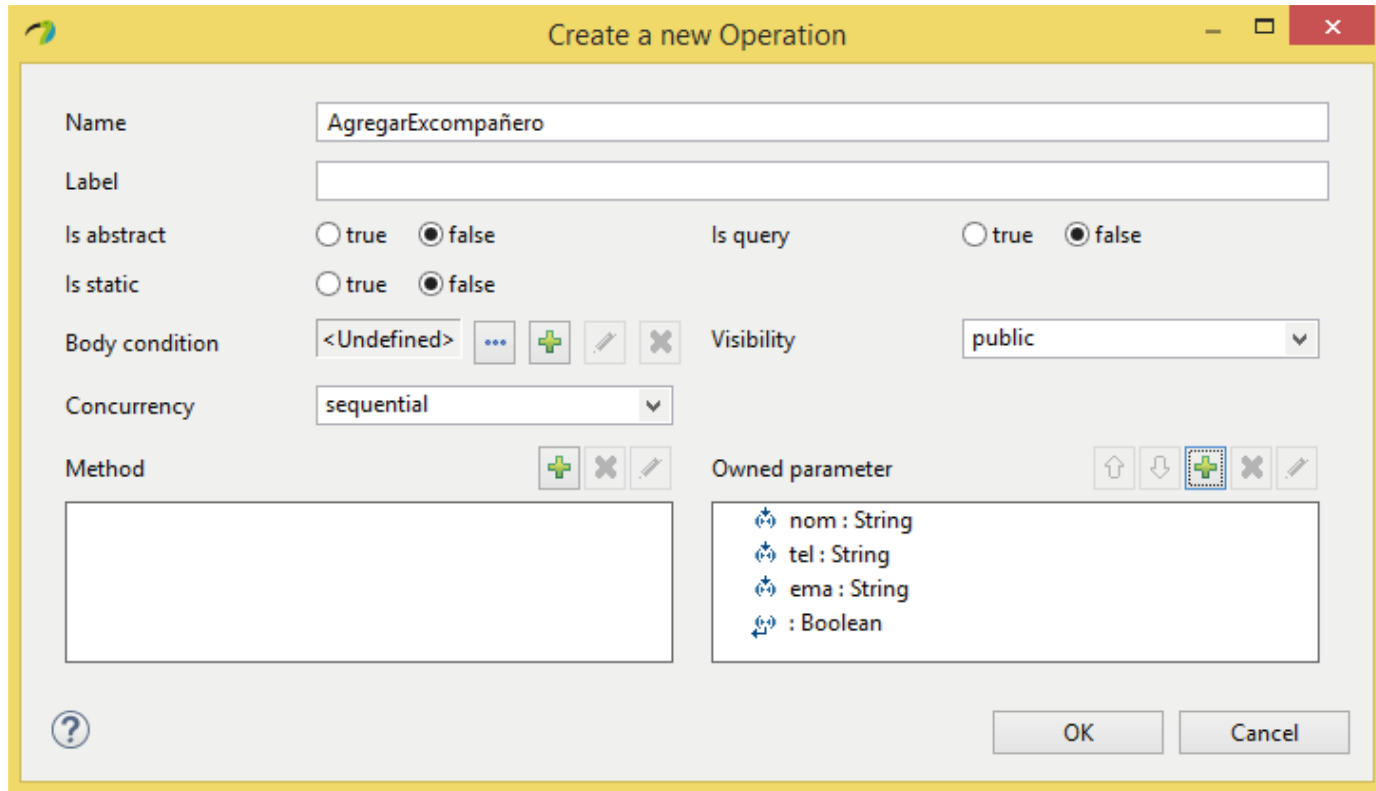


Figura 6.25. Propiedades para el método *AgregarExcompañero* de la clase *Control*: su nombre, visibilidad, sus tres parámetros y el tipo del valor de retorno.

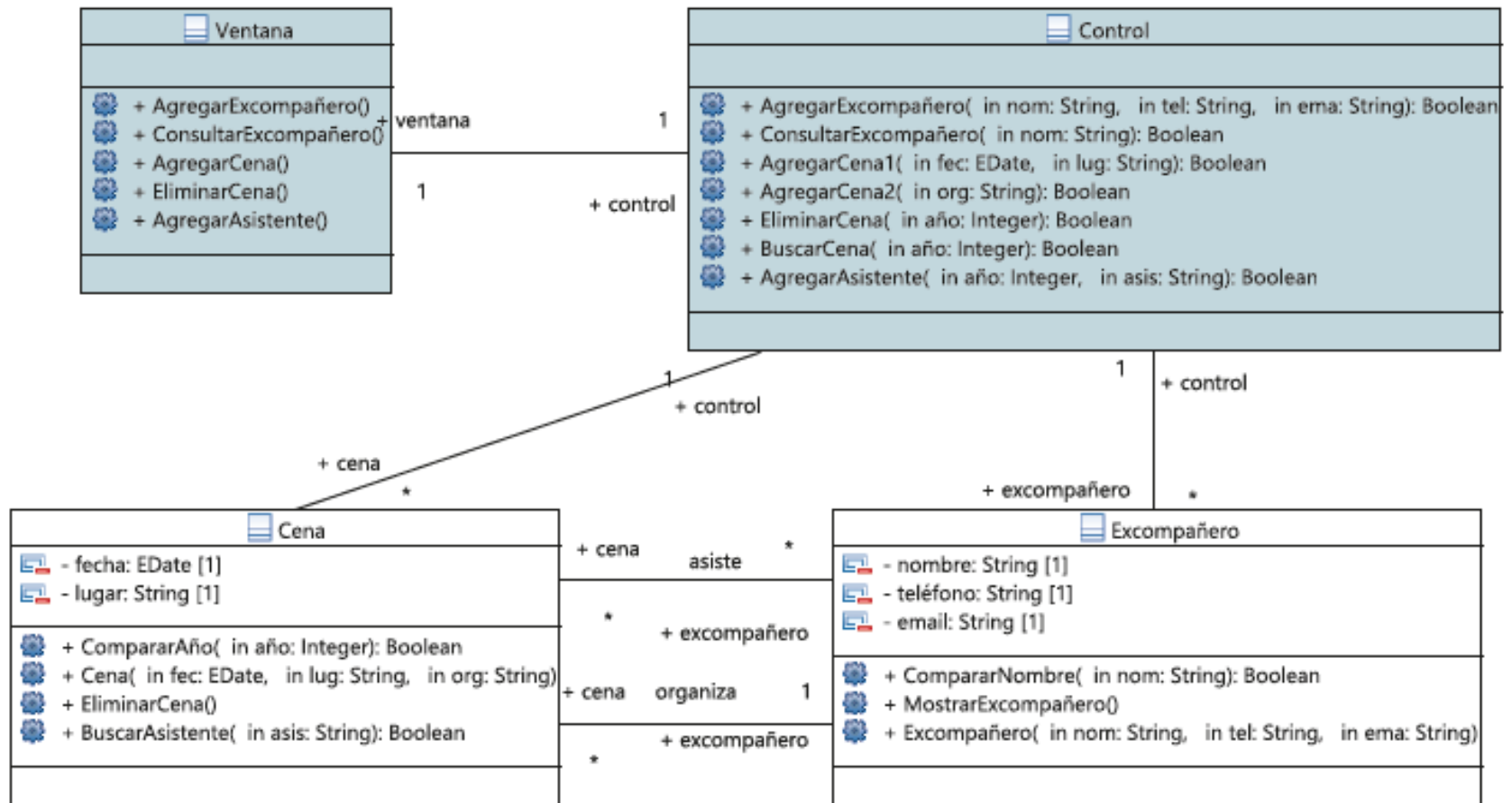


Figura 6.26. Diagrama de clases creado con Papyrus SysML para los casos de uso *Agregar excompañero/a*, *Consultar excompañero/a*, *Agregar cena*, *Eliminar cena* y *Agregar asistente* de la agenda de antiguos compañeros y compañeras.

3.º Se crea el diagrama de secuencia para el caso de uso *Agregar cena*.

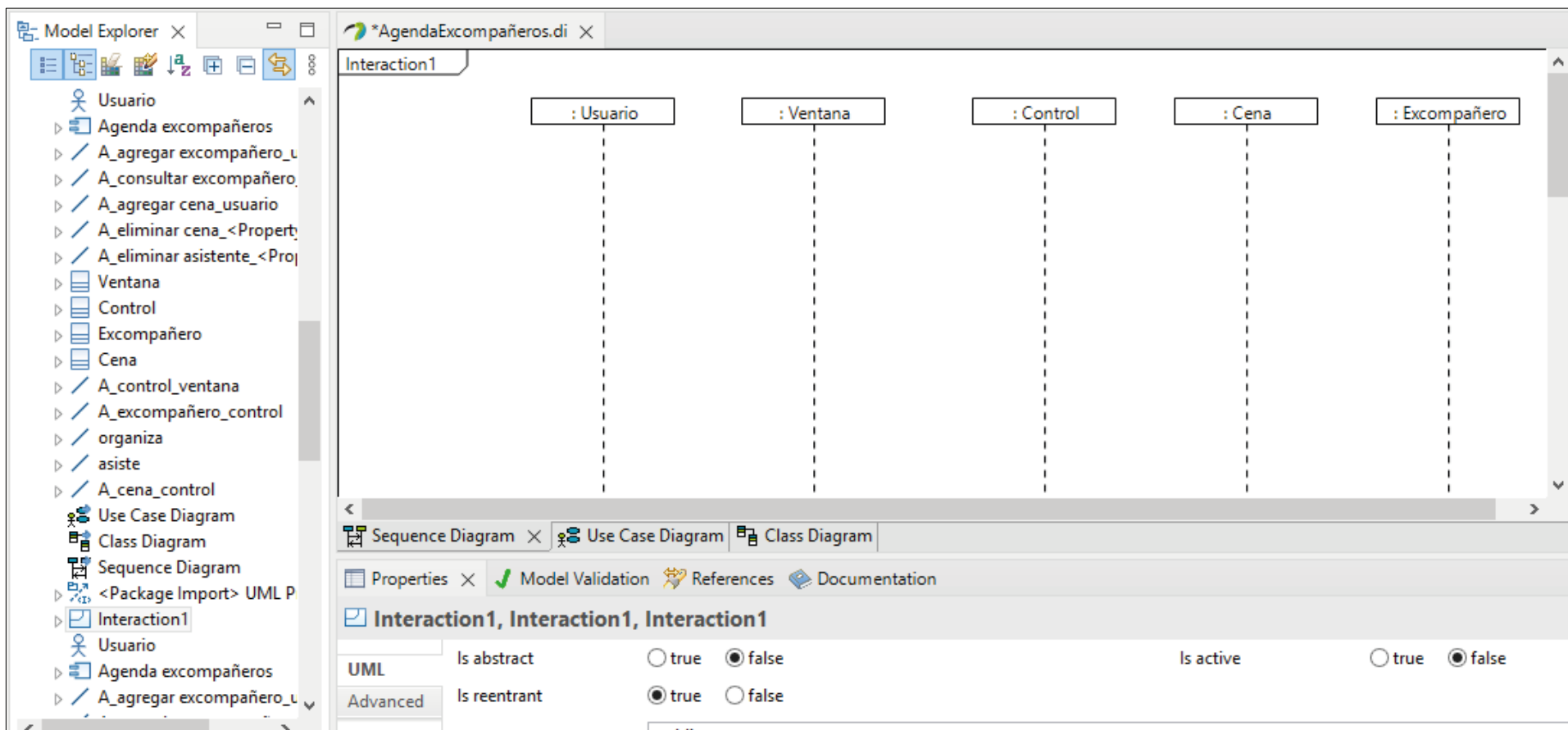


Figura 6.27. Líneas de vida necesarias para el diagrama de secuencia correspondiente al caso de uso *Agregar cena*. Para colocar las líneas de vida, se debe arrastrar el elemento correspondiente de cada línea de vida (actor o clase) desde el explorador del modelo al área de edición del diagrama.

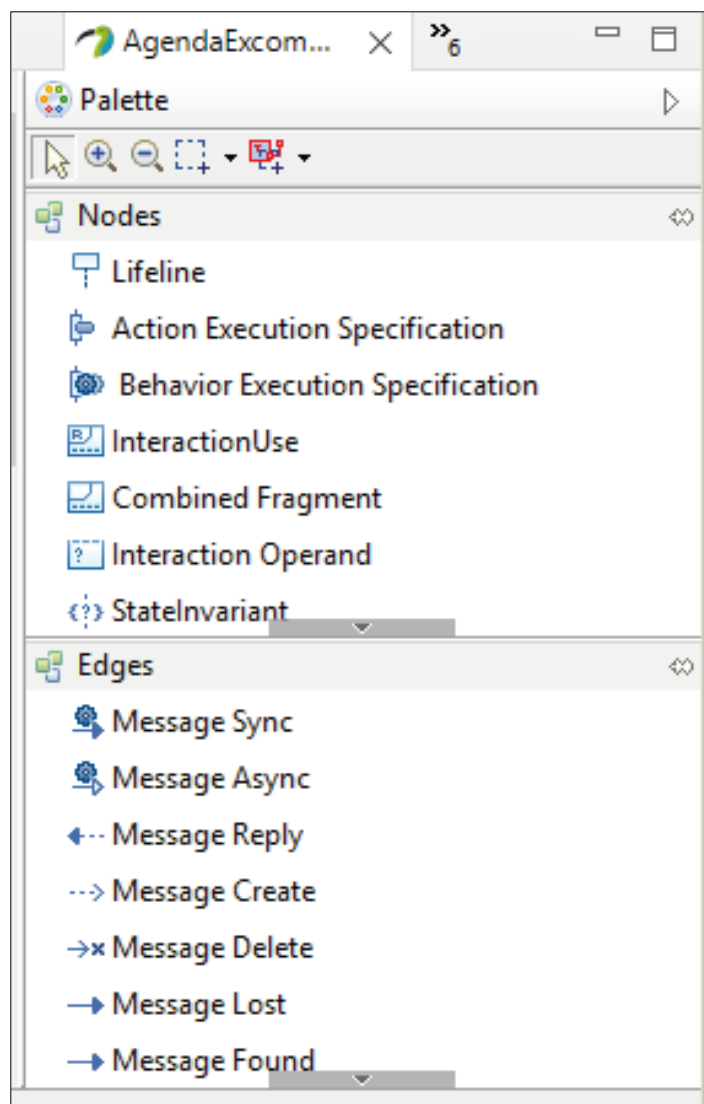


Figura 6.28. Paleta de Papyrus para la elaboración de diagramas de secuencia.

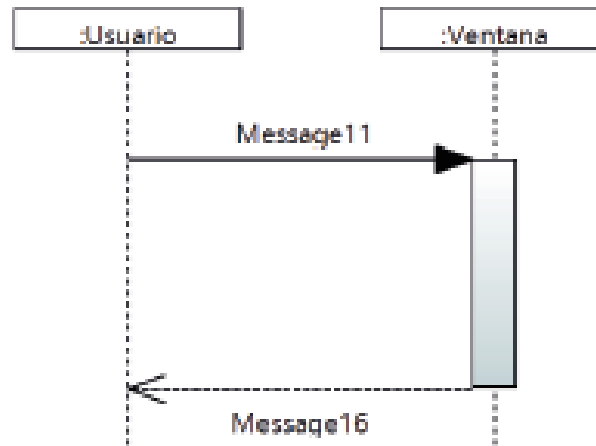


Figura 6.29. Por cada mensaje que se envía en un diagrama de secuencia, Papyrus SysML crea dos mensajes: el mensaje enviado, que en este caso se llama *Message11*, y el mensaje de retorno, llamado *Message16*.

The screenshot shows the 'Properties' window for a selected message named 'Message11'. The window has a sidebar on the left with tabs for 'UML', 'Comments', 'Profile', 'Style', 'Appearance', 'Advance', 'Rulers And Grid', and 'Advanced'. The 'UML' tab is selected. The main area contains the following properties:

- Name:** Message11
- Label:** (empty text field)
- Message sort:** synchCall (dropdown menu)
- Signature:** <Undefined> (with icons for help, add, edit, and delete)
- Argument:** (empty text area with icons for up, down, add, delete, and edit)

Figura 6.30. En la sección *Properties*, tras seleccionar un mensaje, se pueden ver y modificar sus propiedades (su nombre, etiqueta, el método al que se llama, etc.).

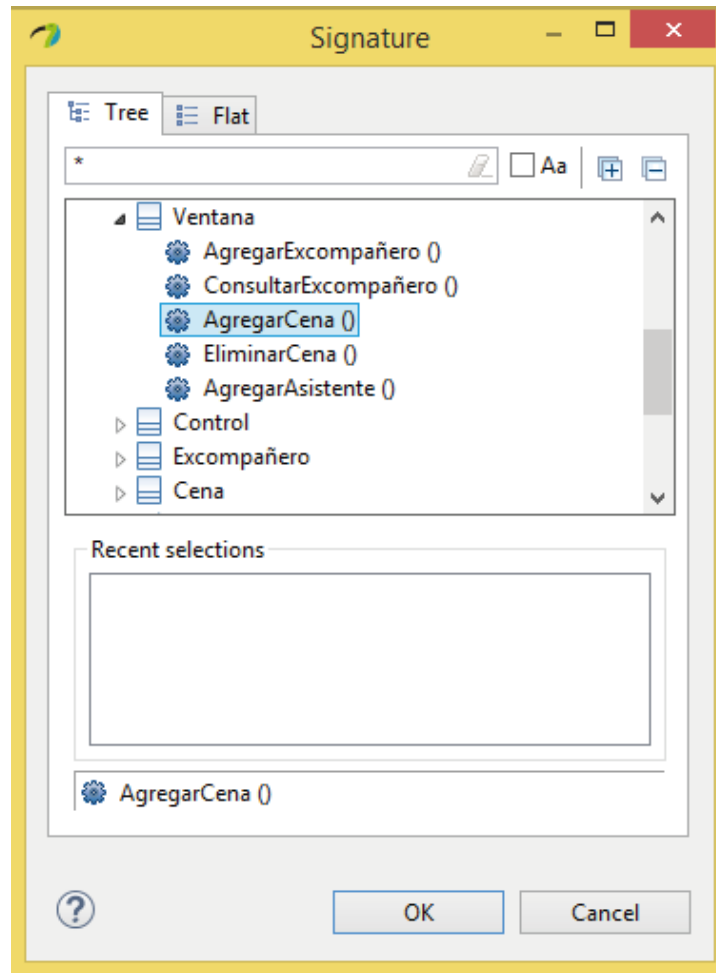


Figura 6.31. Por cada mensaje que se envía en un diagrama de secuencia en Papyrus SysML, se debe seleccionar el método de la clase receptora que se debe ejecutar en respuesta a ese mensaje, asignando un valor al campo *Signature* del mensaje enviado.

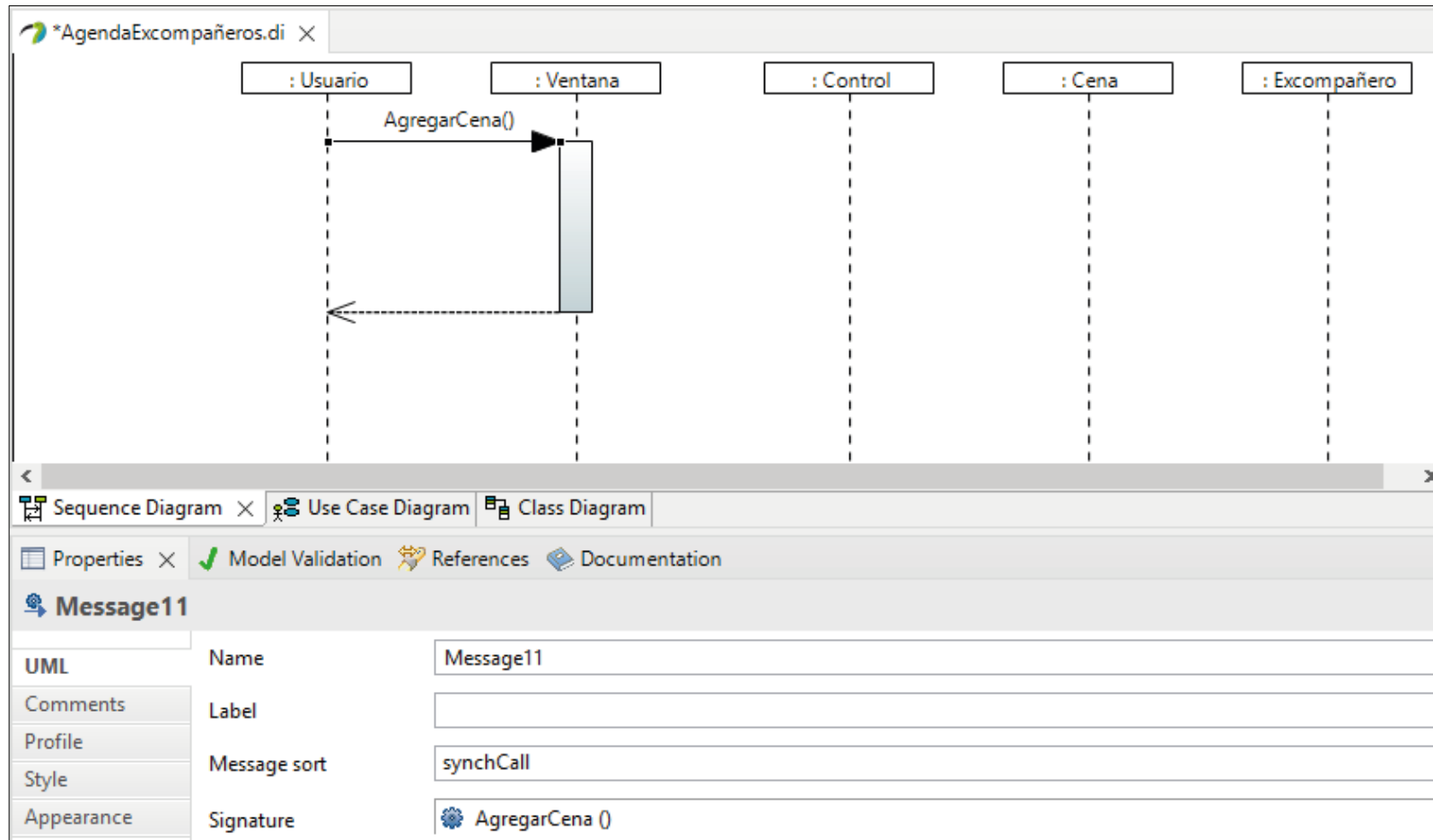


Figura 6.32. Diagrama de secuencia correspondiente al caso de uso *Agregar cena*, en el que se muestra el primer mensaje, que es enviado desde el actor *Usuario* a la clase *Ventana* y solicita la ejecución del método *AgregarCena()*..

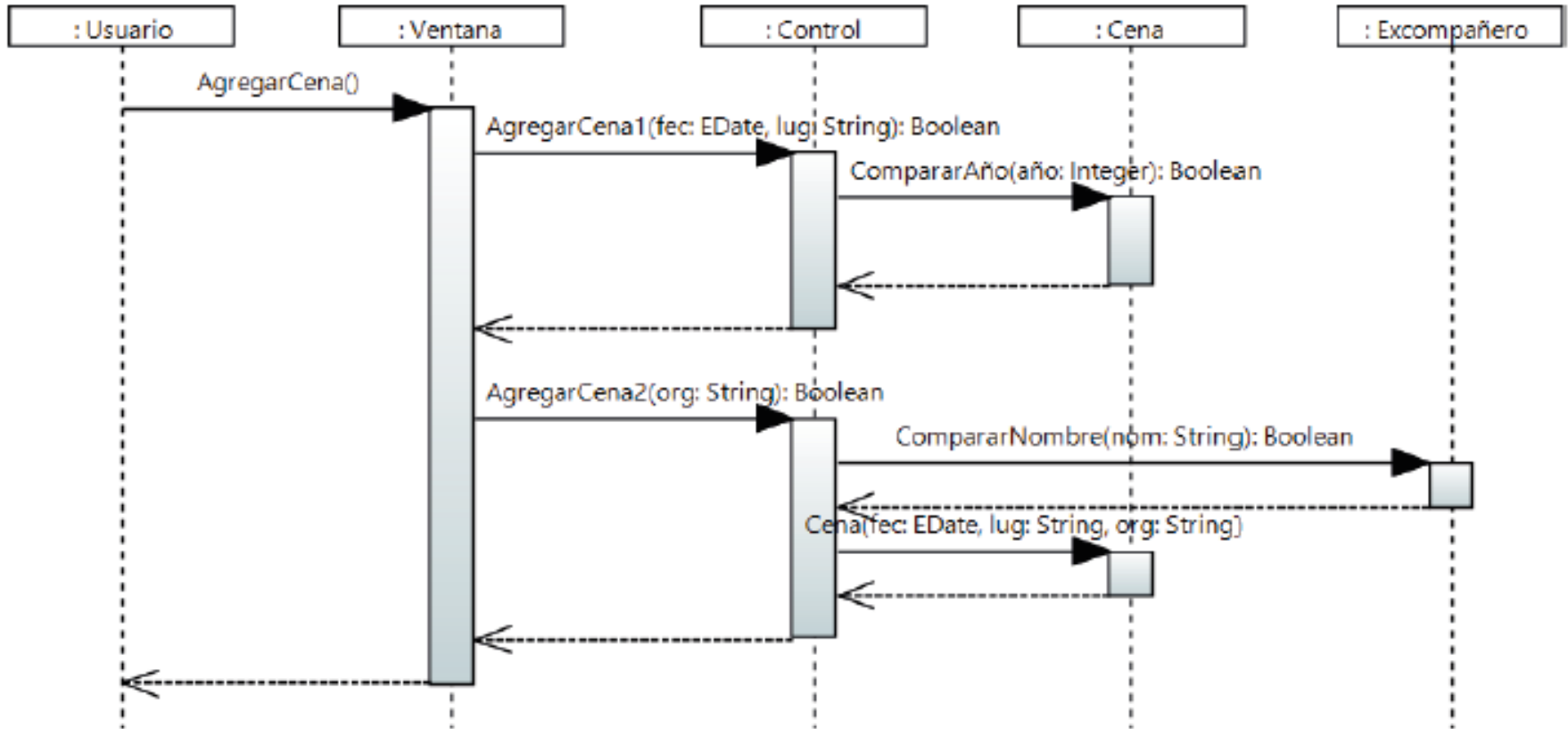

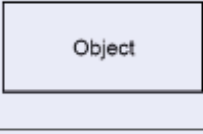




Figura 6.33. Diagrama de secuencia para el caso de uso *Agregar cena* de la agenda de excompañeros y excompañeras realizado con el módulo Papyrus SysML de Eclipse.

Elaboración de diagramas de colaboración con diagrams.net

Tabla 6.5. Diferentes elementos que forman parte de los diagramas de colaboración, su símbolo correspondiente en diagrams.net y la paleta que le corresponde

Elemento	Símbolo	Paleta
Actor	 Actor	UML
Clase u objeto	 Object	UML
Enlace	 Association / Connector / Instance Specification / Property / Connector End	UML
Mensaje	 Message	UML

Elaboración de diagramas de colaboración con Papyrus

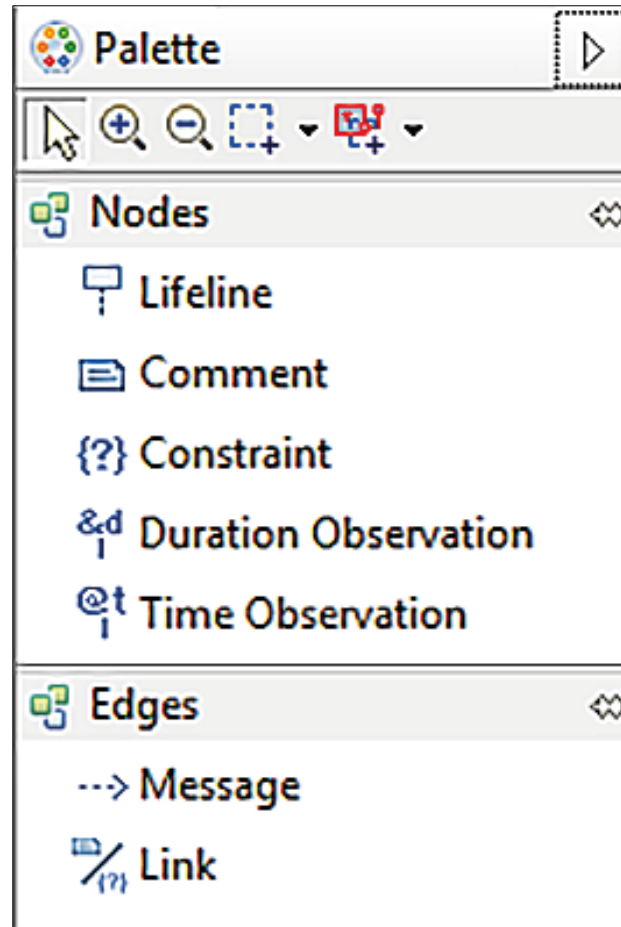


Figura 6.34. Paleta de Papyrus para la elaboración de diagramas de colaboración.

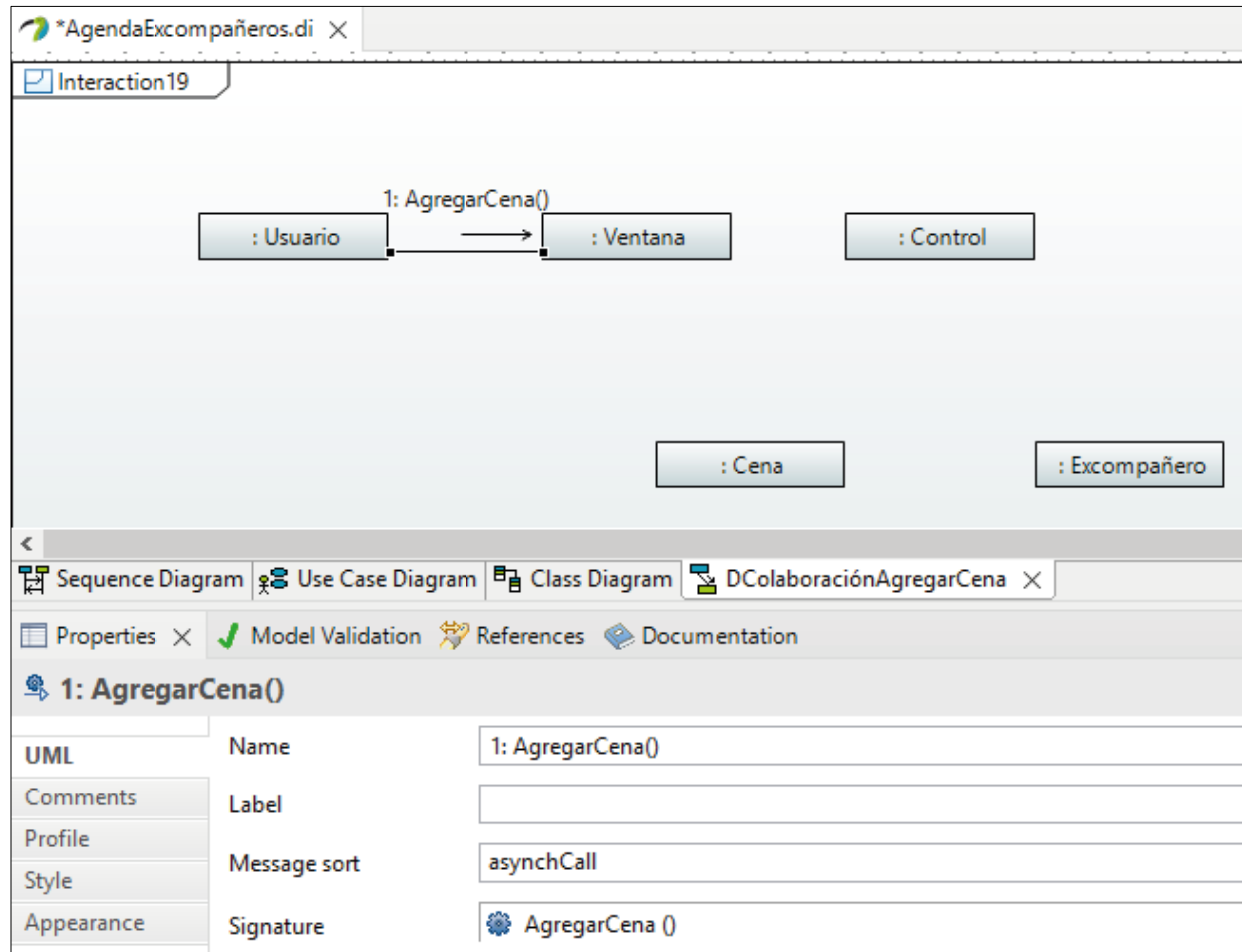


Figura 6.35. Propiedades del mensaje *AgregarCena()*, que es el primero del diagrama de colaboración correspondiente al caso de uso *Agregar cena* de la agenda de excompañeros y excompañeras.

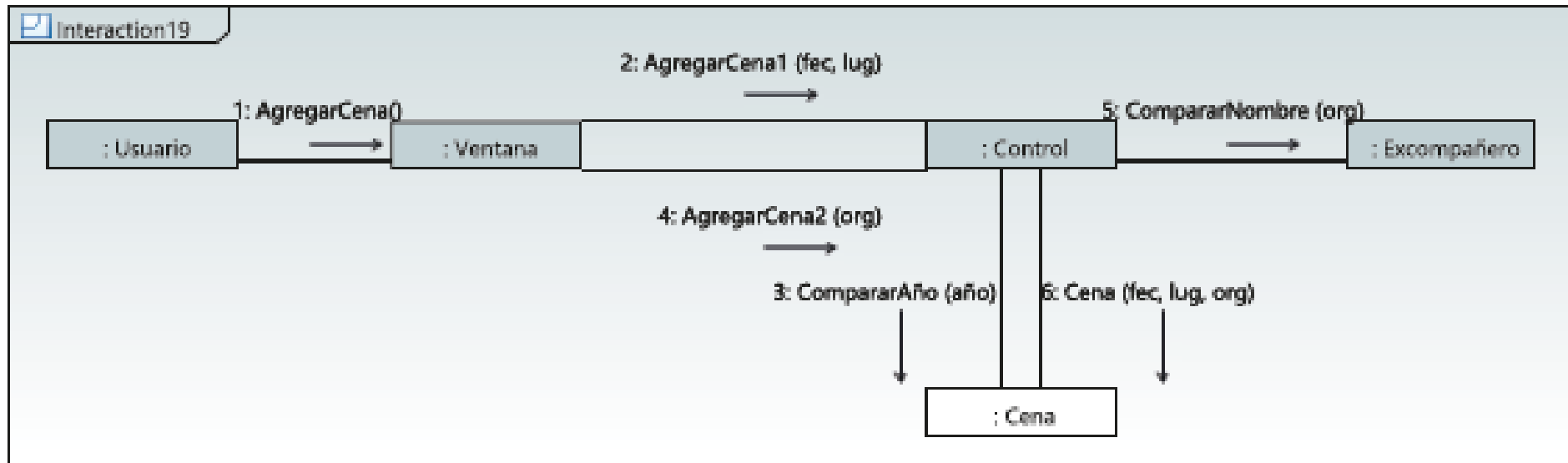


Figura 6.36. Diagrama de colaboración para el caso de uso *Agregar cena* de la agenda de antiguos compañeros y compañeras realizado con el módulo Papyrus SysML de Eclipse.

6.4. Diagramas de estados

- ❑ Muestran los estados (situaciones) por los que pasa un objeto y los cambios de estado consecuencia de eventos.
- ❑ Elementos:

1. Estado.



Figura 6.37. Representación de los estados inicial y final en un diagrama de estados.

2. Evento: suceso que desencadena el paso de un estado a otro.
3. Transición: paso de un estado a otro.

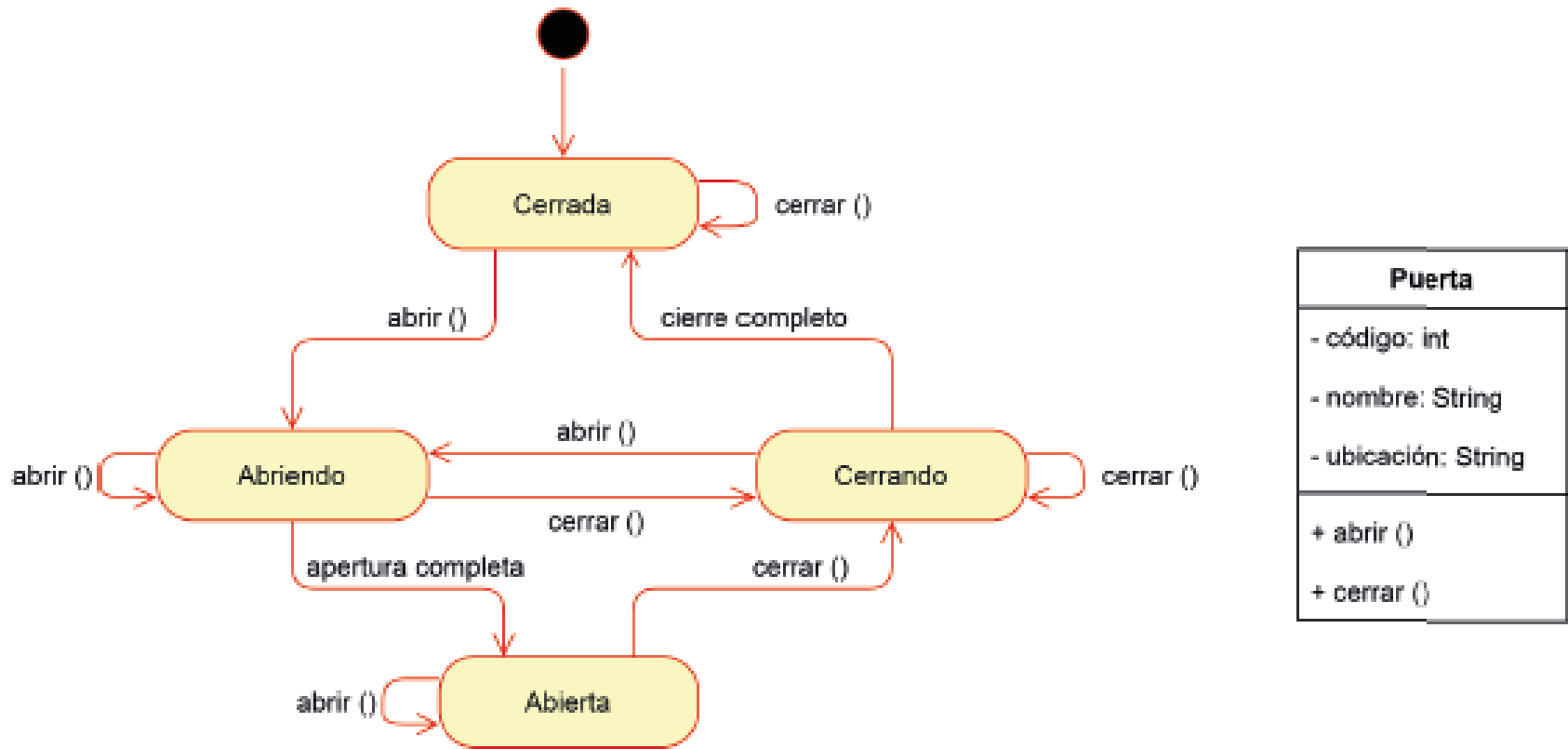


Figura 6.38. Diagrama de estados que representa los estados por los que puede pasar una puerta de garaje.

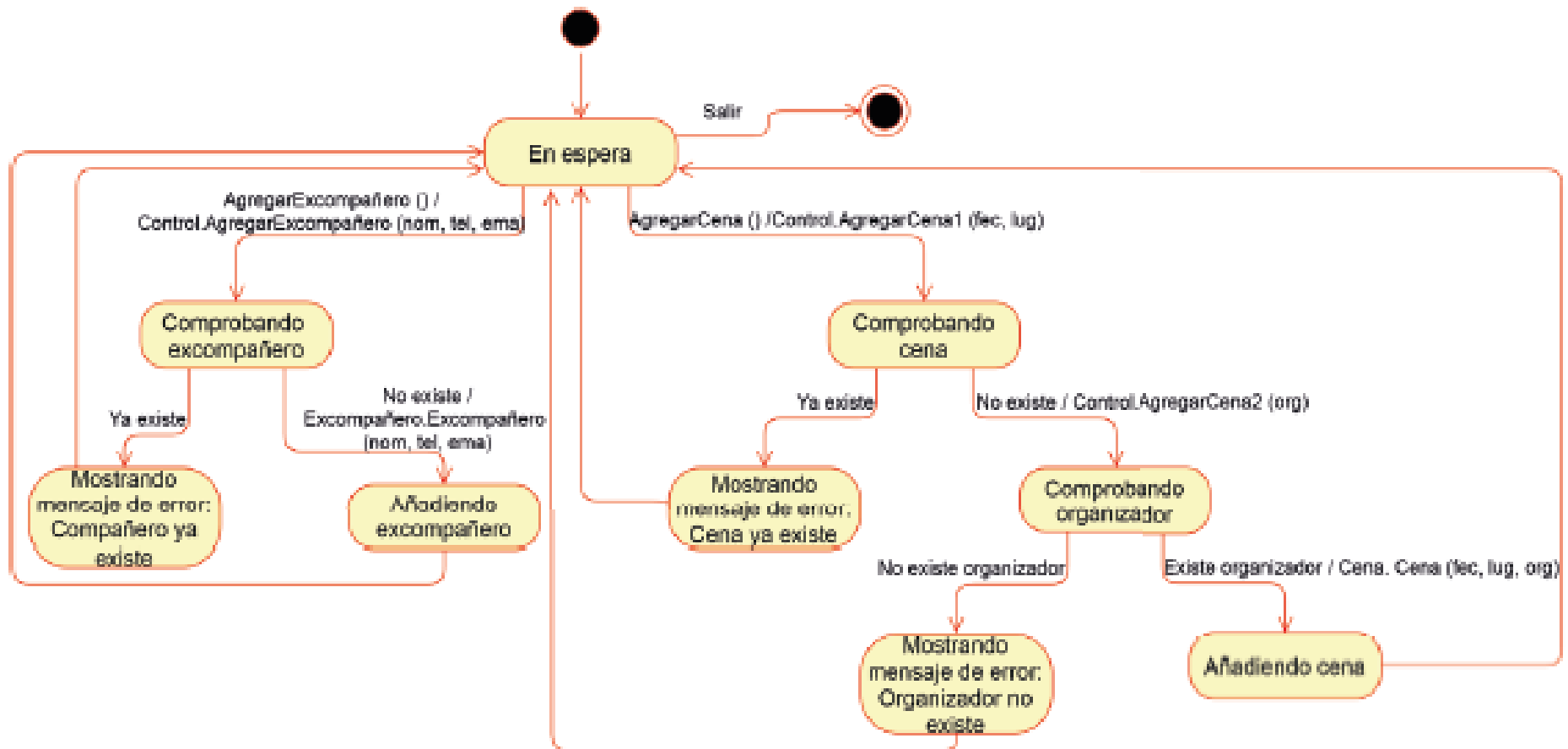


Figura 6.39. Diagrama de estados que representa los estados por los que pasa la clase Ventana en la agenda de excompañeros y excompañeras en relación con los casos de uso *Agregar excompañero/a* y *Agregar cena*.

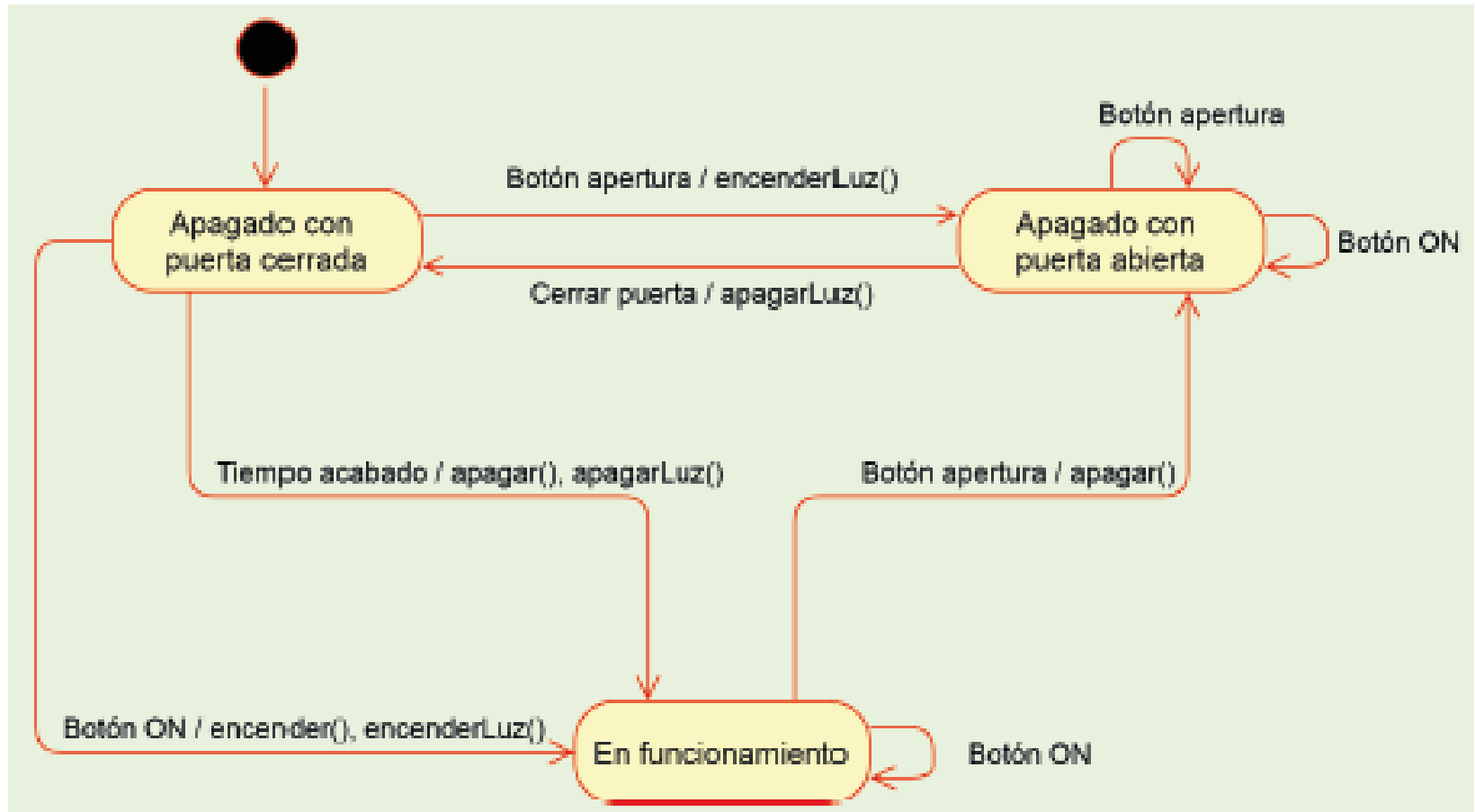


Figura 6.40. Diagrama de estados que representa el funcionamiento de un microondas con dos botones: el de funcionamiento (ON) y el de apertura de la puerta. Este microondas dispone de una luz que debe permanecer encendida mientras está en funcionamiento o si la puerta está abierta.

6.4. Diagramas de actividades

Muestran el flujo de control de las acciones que se realizan en una parte del sistema.

Tabla 6.6. Elementos de un diagrama de actividades



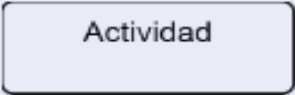

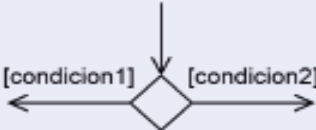
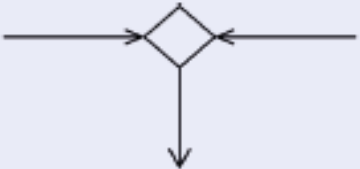
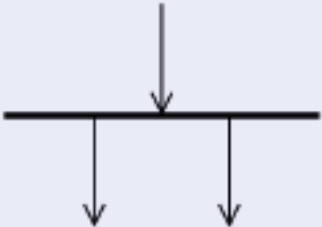
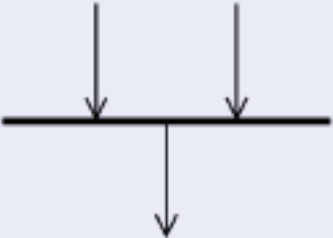
Elemento	Símbolo	Descripción
Nodo de inicio		Representa el inicio del diagrama de actividad.
Nodo de fin		Representa el final del flujo de actividades. Puede que no haya ningún nodo de fin, uno o varios.
Actividad o acción		Representa cada una de las acciones individuales que se ejecutan a lo largo del flujo de trabajo.
Flujo de control		Establece el flujo de control entre las acciones. Después de la acción de la que parte la flecha, se ejecuta la acción a la que llega la flecha.
Bifurcación		El flujo de control se desvía a alguna de las ramas etiquetadas con condiciones. En función de la condición que se cumple, el flujo se dirige por un camino u otro. El rombo de decisión solo puede tener una entrada y puede tener dos o más salidas.

Tabla 6.6. Elementos de un diagrama de actividades (*Continuación*)

Elemento	Símbolo	Descripción
Fusión		<p>El flujo de control de varios caminos se junta en uno solo. El rombo de fusión puede tener varias entradas pero sola una salida.</p>
División		<p>Un flujo de control se separa en dos o más flujos de control concurrentes. Se representa llegando un flujo a una barra de sincronización y saliendo varios. Sirve para indicar que, a partir de determinado punto, se van a ejecutar varias actividades simultáneamente.</p>
Unión		<p>Varios flujos de control concurrentes se unen en un único flujo de control. Se representa llegando a una barra de sincronización varios flujos y saliendo uno solo. Sirve para indicar que a partir de determinado punto finalizan su ejecución varias actividades simultáneas.</p>

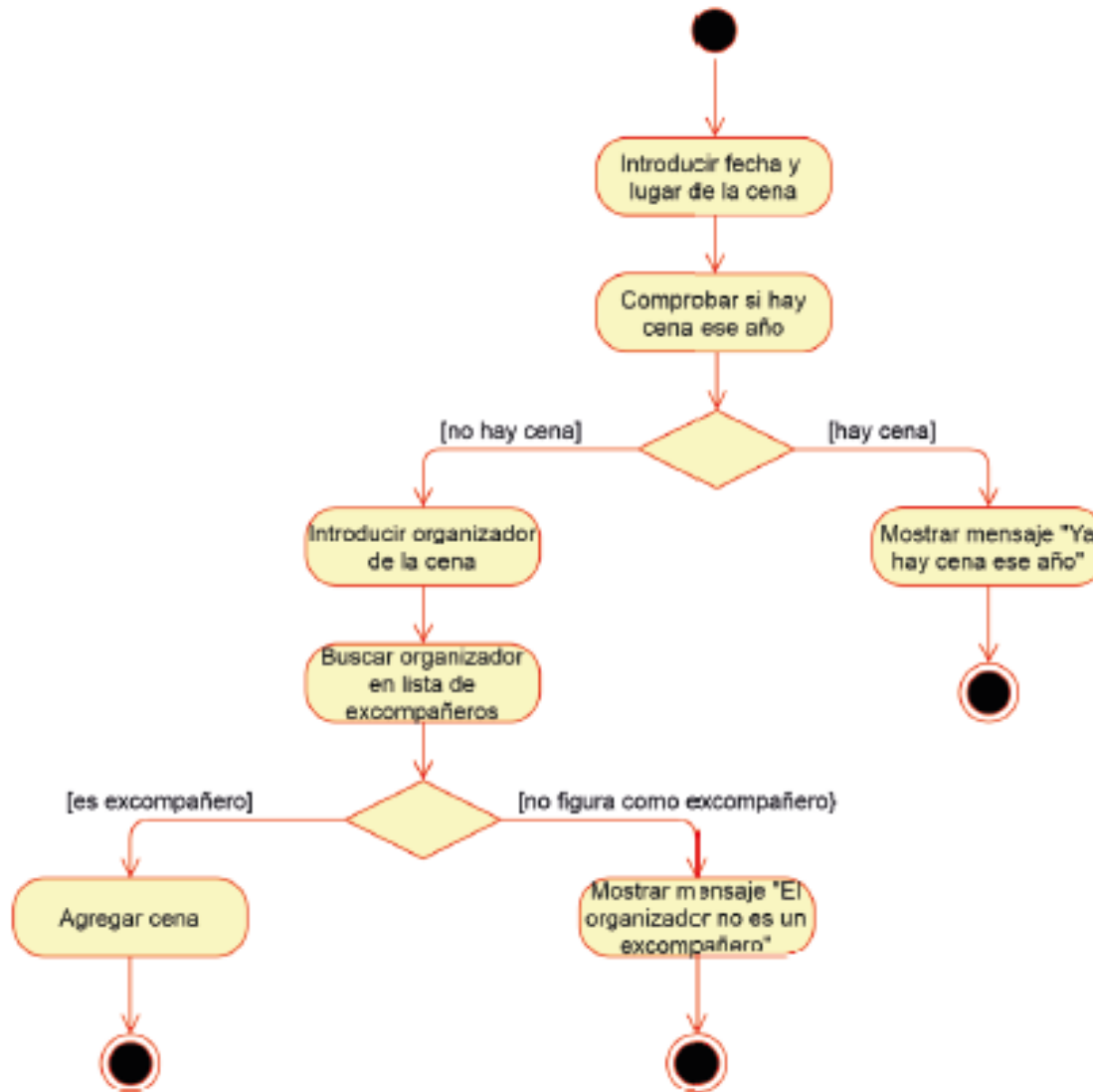


Figura 6.41. Diagrama de actividades que muestra el flujo de control para el caso de uso *Agregar cena*.