



## **UNIVERSIDAD AUSTRAL DE CHILE**

INFO248: Ingeniería de Software

# **Análisis, Diseño y Planificación**

*Estudiante:*

Ignacio Castro

Enzo Meneses

*Profesores:*

Dr. Raimundo Vega

Dr. Cristian Olivares-Rodríguez

VALDIVIA, CHILE

2020

# Casos de Uso

## Agregar reserva:

**Caso de uso:** Agregar reserva

**Actores:** Administrador, cliente

**Propósito:** Agregar una reserva en sistema a nombre del cliente.

**Precondición:** El administrador se ha identificado y autenticado.

**Resumen:** Un cliente se contacta con el administrador con la intención de reservar una cabaña. El administrador revisa la disponibilidad, y en caso de haber, ingresa los datos del cliente y la reserva. Al terminar la operación el cliente tiene una reserva de cabañas.

**Tipo:** Primario

**Curso normal de los eventos:**

Acción de los actores	Acción del sistema
1) El caso comienza cuando un cliente se contacta con el administrador para realizar una reserva	
2) El administrador ingresa los datos de la reserva.	3) Verifica si hay disponibilidad e indica el precio.
4) El administrador selecciona el cliente.	5) Indica que la reserva se guardó correctamente. Envía correo con datos de la reserva al cliente.
6) El cliente termina la conversación con una reserva.	

## Cursos alternativos:

Línea 3: No hay disponibilidad. Se le pregunta al cliente si le acomodan otras fechas. Dependiendo de la respuesta se vuelve a la línea 2 o se cancela la operación.

Línea 3: El cliente no está conforme con el precio. Se conversa la posibilidad de un descuento. Si no es suficiente se cancela la operación.

Línea 4: El cliente no está registrado en el sistema. Se procede al caso de uso Agregar cliente.

Línea 5: No se guarda la reserva, se repite el proceso.

Línea 6: El cliente no recibe el correo. Se verifica que el correo sea correcto. Si no lo es, se pasa al caso de uso de modificar cliente. Si lo es se pasa al caso de uso de Enviar correo

## **Modificar Cliente:**

**Caso de uso:** Modificar cliente

**Actores:** Administrador, cliente.

**Propósito:** Actualizar o corregir los datos del cliente.

**Precondición:** El administrador se ha identificado y autenticado.

El cliente está registrado en la base de datos.

**Resumen:** El cliente o el administrador se dan cuenta que los datos del cliente guardados están desactualizados o erróneos. Por lo que se comunican entre sí para arreglarlos o actualizarlos.

**Tipo:** Primario

**Curso normal de los eventos:**

Acción de los actores	Acción del sistema
1) El cliente o el administrador se dan cuenta que los datos guardados están desactualizados o erróneos, y se comunican para obtener los datos correctos.	
2) El administrador busca el registro del cliente.	3) Muestra los datos guardados del cliente.
4) El administrador modifica los datos del cliente.	5) Indica que la reserva se guardó correctamente.
6) La operación termina con los datos guardados correctamente.	

**Cursos alternativos:**

Línea 3: El cliente no está registrado en el sistema. Se pasa al caso de uso Agregar reserva o Agregar usuario.

Línea 5: No se guarda la reserva, se repite el proceso.

## ***Agregar Cliente:***

**Caso de uso:** Agregar cliente

**Actores:** Administrador, cliente.

**Propósito:** Agregar un cliente a la base de datos.

**Precondición:** El administrador se ha identificado y autenticado.

**Resumen:** Un cliente se contacta con el administrador, porque tiene pensado realizar una reserva, pero no tiene claro los detalles de esta, por lo que se decide agregar sus datos para facilitar el proceso cuando realice la reserva.

**Tipo:** Primario

**Curso normal de los eventos:**

Acción de los actores	Acción del sistema
1) El caso comienza cuando un cliente se contacta con el administrador para estar registrado en el sistema de las cabañas	
2) El administrador busca el registro del cliente.	3) Indica que el cliente no está registrado.
4) El administrador ingresa los datos del cliente.	5) Indica que el cliente fue registrado correctamente.
6) El cliente queda en la base de datos del sistema.	

**Cursos alternativos:**

Línea 3: El cliente ya está registrado. Se termina la operación.

## **Modificar Reserva:**

**Caso de uso:** Modificar reserva

**Actores:** Administrador, cliente.

**Propósito:** Modificar una reserva.

**Precondición:** El administrador se ha identificado y autenticado.

La reserva está registrada en la base de datos.

**Resumen:** El cliente se comunica con el administrador para modificar una reserva. El administrador busca la reserva y modifica los datos de esta.

**Tipo:** Primario

**Curso normal de los eventos:**

Acción de los actores	Acción del sistema
1) 1) El caso comienza cuando un cliente se contacta con el administrador para modificar una reserva.	
2) El administrador busca el cliente.	3) Muestra el registro del cliente.
4) El administrador selecciona la reserva.	5) Muestra los datos guardados de la reserva.
6) El administrador modifica los datos de la reserva.	7) Verifica si hay disponibilidad e indica el precio.
8) El administrador completa la modificación.	9) Indica que la reserva se guardó correctamente. Envía correo con nuevos datos de la reserva.
10) El cliente tiene una reserva modificada.	

**Cursos alternativos:**

Línea 3: El cliente no está en el sistema. Se pasa al caso de uso Agregar cliente.

Línea 5: La reserva no está en el sistema. Se pasa al caso de uso Agregar reserva.

Línea 7: No hay disponibilidad. Se le pregunta al cliente si le acomodan otras fechas. Dependiendo de la respuesta se vuelve a la línea 2 o se cancela la operación.

Línea 7: El cliente no está conforme con el precio. Se conversa la posibilidad de un descuento. Si no es suficiente se cancela la operación.

Línea 9: No se guarda la reserva, se repite el proceso.

Línea 10: El cliente no recibe el correo. Se verifica que el correo sea correcto. Si no lo es, se pasa al caso de uso de modificar cliente. Si lo es se pasa al caso de uso de Enviar correo

### ***Modificar Cabaña:***

**Caso de uso:** Modificar cabaña

**Actores:** Administrador.

**Propósito:** Modificar los datos de una cabaña.

**Precondición:** El administrador se ha identificado y autenticado.

**Resumen:** Por alguna razón es necesario modificar una cabañas (cambiar precio, deshabilitarla, etc). El administrador busca la cabaña a modificar y realiza los cambios necesarios.

**Tipo:** Primario

**Curso normal de los eventos:** Cuando es necesario modificar una cabaña el administrador la busca en el sistema, y realiza los cambios. En caso generar conflictos con reservas aparece un mensaje de atención y para que el administrador decida si efectuar los cambios o no.

Acción de los actores	Acción del sistema
1) El caso comienza cuando se debe realizar una modificación a los datos de una cabaña.	
2) El administrador busca la cabaña a modificar	3) Despliega los datos guardados de la cabaña.
4) El administrador modifica los datos necesarios y los guarda.	5) Indica que los cambios se guardaron correctamente.
6) Termina el proceso con los datos de la cabaña modificados.	

### **Cursos alternativos:**

Línea 5: Los cambios generan conflictos con alguna reserva. Se muestra un mensaje de Warning con las opciones de confirmar o cancelar los cambios. Dependiendo de la respuesta se continúa normalmente o se cancela la operación.

Línea 5: No se guarda el cambio, se repite el proceso.

## ***Eliminar Reserva:***

**Caso de uso:** Eliminar reserva.

**Actores:** Administrador, cliente.

**Propósito:** Eliminar una reserva de la base de datos.

**Precondición:** El administrador se ha identificado y autenticado.

La reserva está registrada en la base de datos.

**Resumen:** Un cliente se contacta con el administrador para cancelar una reserva. El administrador busca la reserva en el sistema y la elimina.

**Tipo:** Primario

**Curso normal de los eventos:**

<b>Acción de los actores</b>	<b>Acción del sistema</b>
1) El caso comienza cuando el cliente se comunica con el administrador para cancelar una reserva	
2) El administrador busca la reserva en el sistema.	3) Despliega los datos guardados de la reserva.
4) El administrador elimina la reserva.	5) Indica que la reserva fue eliminada.
6) Termina el proceso y el cliente ya no tiene la reserva que eliminó.	

**Cursos alternativos:**

Línea 3: La reserva no se encuentra en el sistema. Se termina el proceso.

Línea 5: No se elimina la reserva correctamente. Se repite el proceso.

## ***Eliminar Cliente:***

**Caso de uso:** Eliminar cliente.

**Actores:** Administrador, cliente.

**Propósito:** Eliminar una reserva de la base de datos.

**Precondición:** El administrador se ha identificado y autenticado.

La reserva está registrada en la base de datos.

**Resumen:** Un cliente se contacta con el administrador porque no quiere que sus datos estén guardados en el sistema. El administrador busca al cliente en el sistema y la elimina.

**Tipo:** Primario

**Curso normal de los eventos:**

<b>Acción de los actores</b>	<b>Acción del sistema</b>
1) El caso comienza cuando el cliente se comunica con el administrador para eliminar sus datos del sistema.	
2) El administrador busca el cliente en el sistema.	3) Despliega los datos guardados de clientel.
4) El administrador elimina el cliente.	
6) Termina el proceso y los datos del cliente ya no están en el sistema.	5) Indica que el cliente fue eliminado.

**Cursos alternativos:**

Línea 3: el cliente no está registrado en el sistema. Se termina la operación.

Línea 5: No se elimina el cliente correctamente. Se repite el proceso.



## ***Enviar Correos:***

**Caso de uso:** Enviar correos.

**Actores:** Administrador.

**Propósito:** Enviar correos automatizados a los clientes.

**Precondición:** El administrador se ha identificado y autenticado.

**Resumen:** El administrador envía un correo automatizado a los clientes con los datos de la reserva y de transferencia o como recordatorio para realizar el pago de esta.

**Tipo:** Primario

**Curso normal de los eventos:**

Acción de los actores	Acción del sistema
1) El caso comienza cuando el administrador necesita enviar un correo automatizado a un cliente.	
2) El administrador busca el cliente en el sistema.	3) Despliega los datos guardados del cliente.
4) El administrador envía un correo.	5) Indica que el correo fue enviado.
6) Termina el proceso y el correo fue enviado al cliente.	

### **Cursos alternativos:**

Línea 3: El cliente no está registrado en el sistema. Se cancela la operación. El administrador debe comunicarse con el cliente en caso de ser necesario.

Línea 5: No se envía el correo. Se repite la operación.

## ***Consultas Redes Sociales:***

**Caso de uso:** Consultas redes sociales.

**Actores:** Administrador.

**Propósito:** Responder a consultas de clientes realizadas por redes sociales..

**Precondición:** El administrador se ha identificado y autenticado.

**Resumen:** El sistema avisa al administrador que hay una consulta de un cliente. El administrador responde la consulta y la respuesta es enviada al cliente.

**Tipo:** Secundario

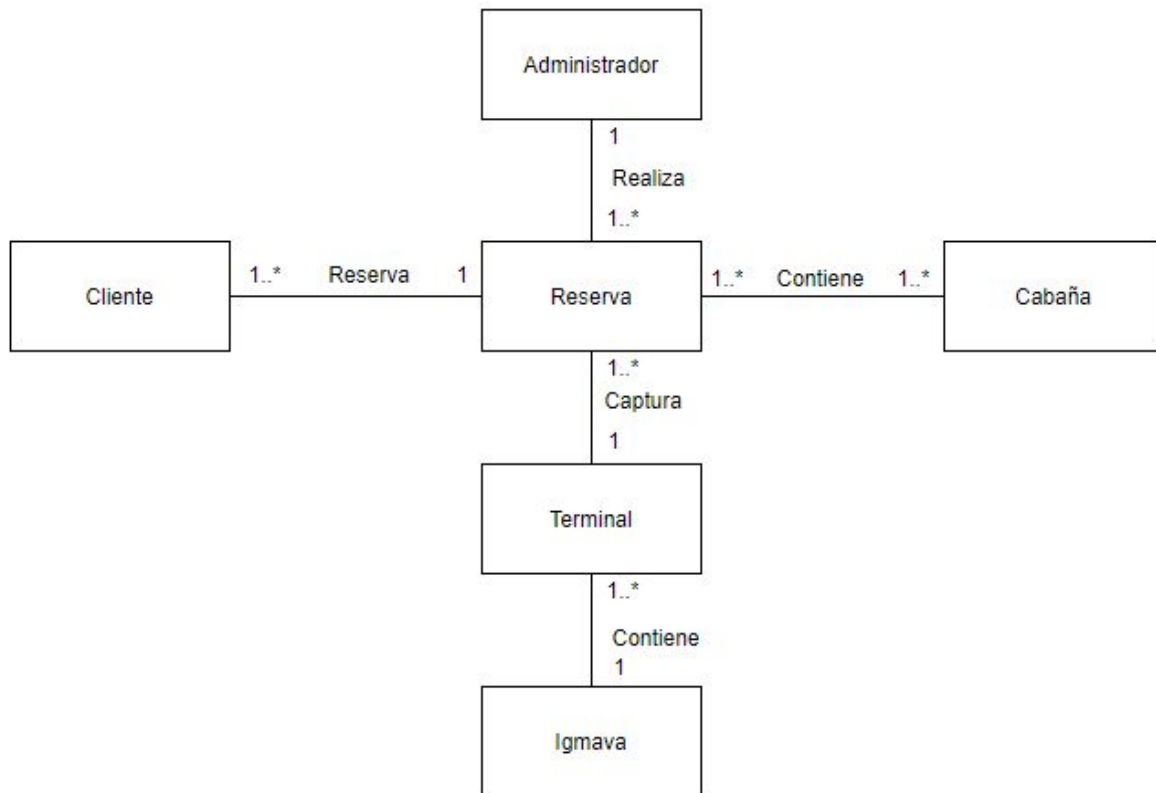
**Curso normal de los eventos:**

Acción de los actores	Acción del sistema
2) El administrador responde la pregunta	1) El sistema avisa al administrador que hay una consulta por responder 3) Envía respuesta al cliente.

**Cursos alternativos:**

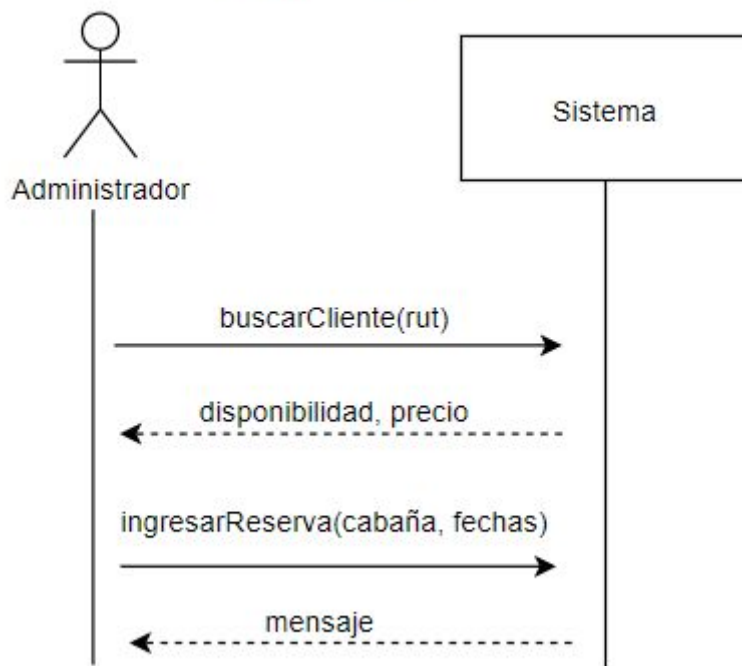
Línea 3: No se envía la respuesta correctamente. Se repite la operación.

## Modelo Conceptual

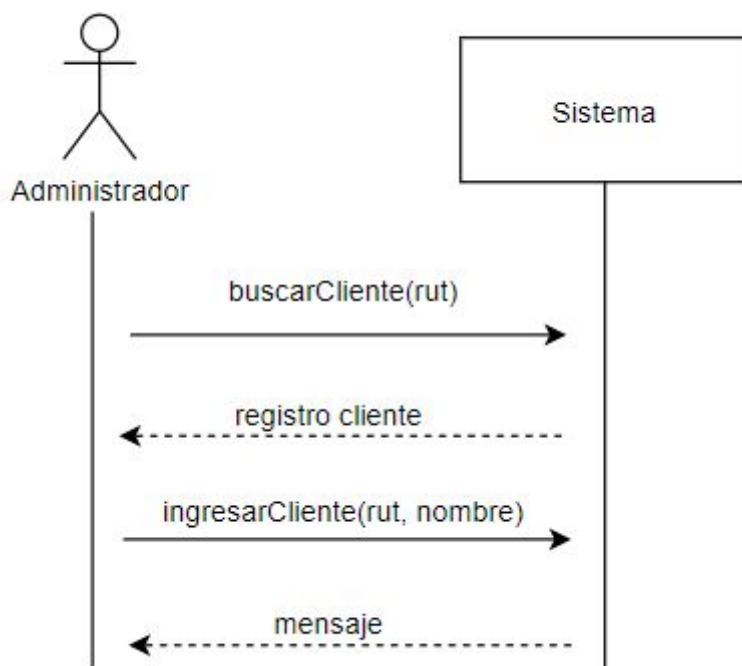


## Diagramas de secuencia del sistema

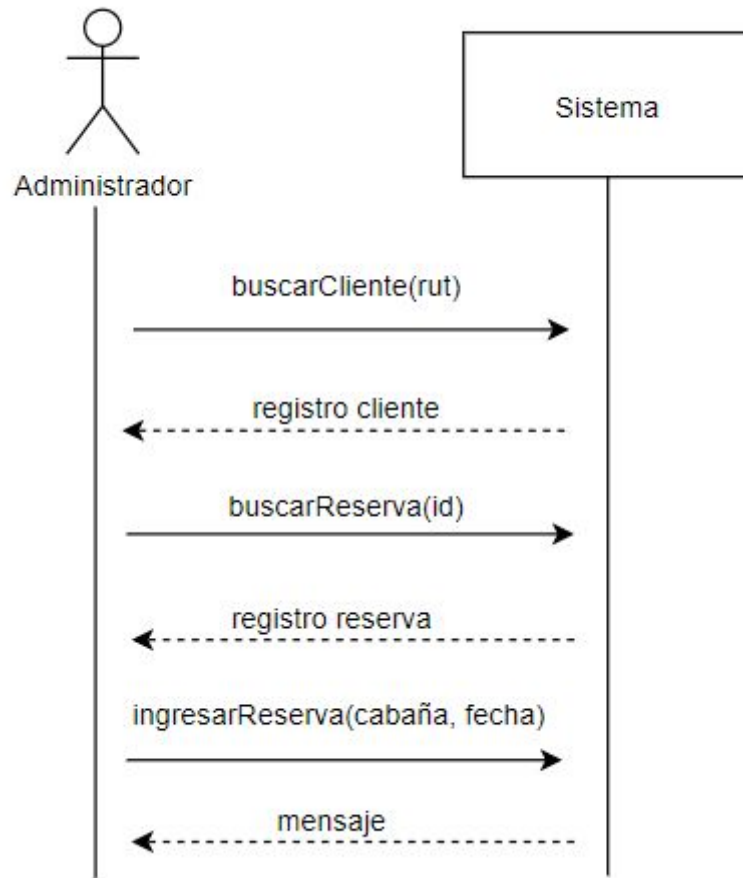
### Agregar reserva



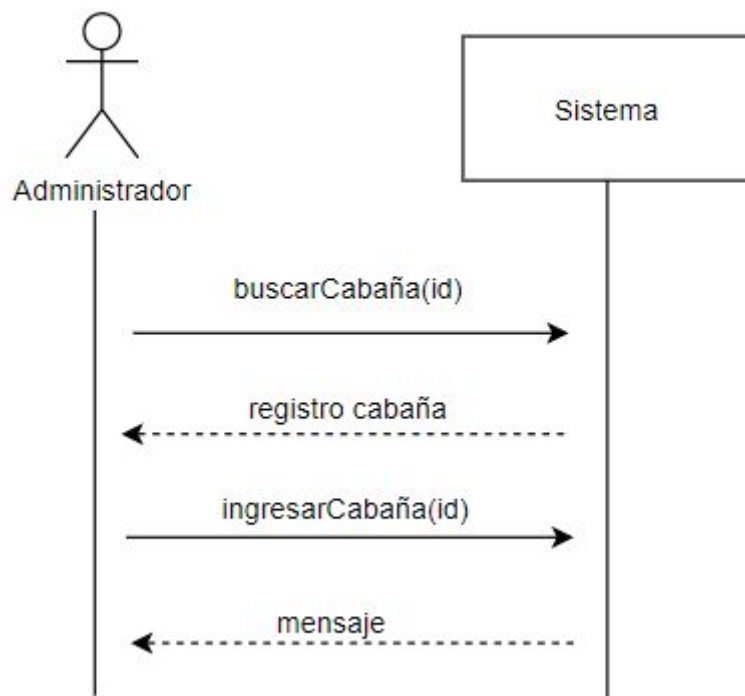
### Agregar/modificar cliente



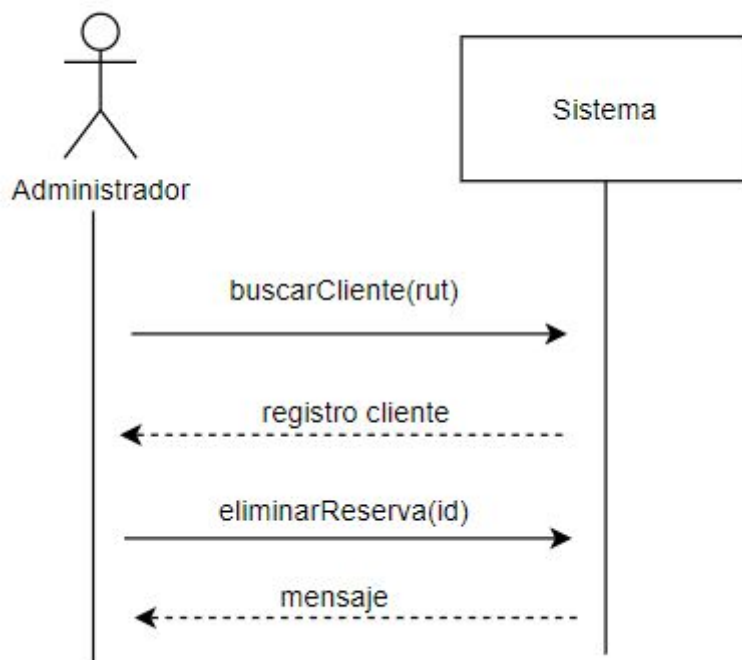
## Modificar Reserva



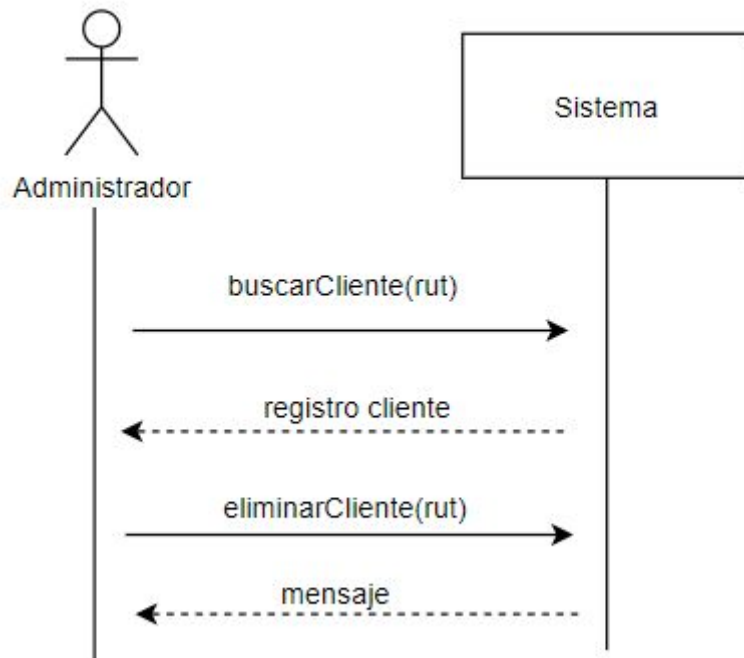
### Modificar cabaña



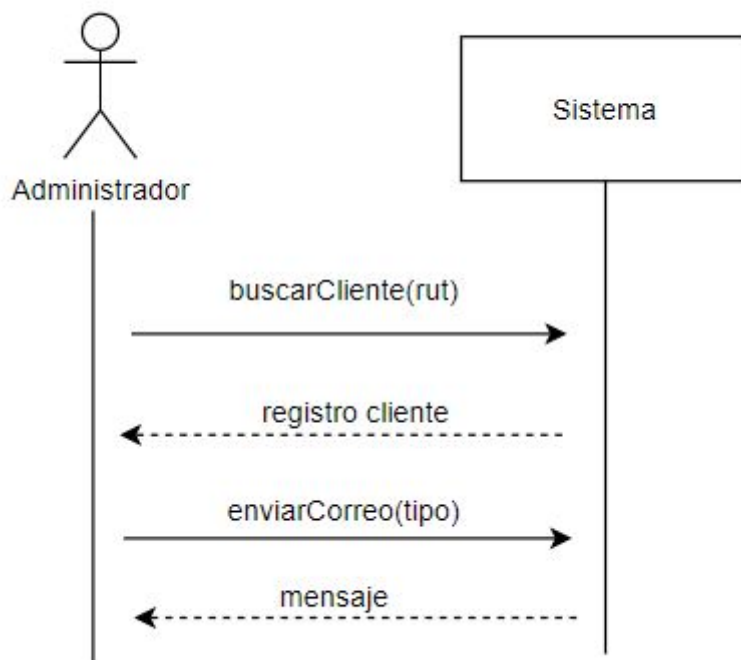
### Eliminar reserva

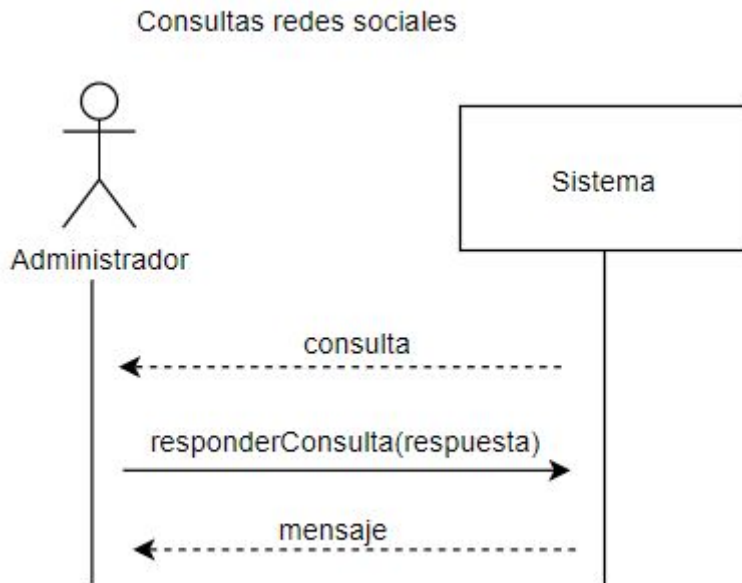


### Eliminar cliente



### Enviar correo





## Contratos

### ***buscarCliente(rut):***

**Nombre:** buscarCliente(rut)

**Responsabilidades:** Buscar cliente en la base de datos.  
Desplegar información del cliente.

**Excepciones:** El rut no es válido.

**Referencias cruzadas:** Caso de uso: Agregar reserva  
Caso de uso: Agregar cliente  
Caso de uso Modificar cliente  
Caso de uso: Modificar reserva  
Caso de uso: Eliminar reserva  
Caso de uso: Eliminar cliente  
Caso de uso: Enviar correo

**Precondiciones:**

**Poscondiciones:**

Se creó una instancia de Cliente.



### ***ingresarReserva(cabañas, fechas):***

**Nombre:** ingresarReserva(cabañas, fechas)

**Responsabilidades:** Capturar reserva de cabañas y agregarla al sistema.  
Desplegar disponibilidad y precio de la reserva.

**Excepciones:** No hay disponibilidad de cabañas.

**Referencias cruzadas:** Caso de uso: Agregar reserva  
Caso de uso: Modificar reserva

**Precondiciones:**

**Poscondiciones:**

Se creó la Reserva.

Se asoció al Terminal.

Se asoció la Reserva al Cliente.

### ***ingresarCliente(rut, nombre):***

**Nombre:** ingresarCliente(rut, nombre)

**Responsabilidades:** Capturar cliente y agregarlo al sistema.

**Excepciones:**

**Referencias cruzadas:** Caso de uso: Agregar cliente  
Caso de uso: Modificar cliente

**Precondiciones:**

**Poscondiciones:**

Si es un cliente nuevo, se creó el Cliente.

Se creó una instancia de Cliente.

Se cambiaron los atributos de Cliente

Se guardó la instancia del Cliente en la base de datos.

### ***buscarReserva(id):***

**Nombre:** buscarReserva(id)

**Responsabilidades:** Buscar reserva en la base de datos.  
Desplegar información de la reserva.

**Excepciones:**

**Referencias cruzadas:** Caso de uso: Modificar reserva  
Caso de uso: Eliminar reserva

**Precondiciones:**

**Poscondiciones:**

Se creó una instancia de Reserva..

### ***buscarCabaña(id):***

**Nombre:** buscarCabaña(id)

**Responsabilidades:** Buscar cabaña en la base de datos.  
Desplegar información de la cabaña.

**Excepciones:**

**Referencias cruzadas:** Caso de uso: Modificar reserva

**Precondiciones:**

**Poscondiciones:**

Se creó una instancia de Cabaña.

### ***ingresarCabaña(id):***

**Nombre:** ingresarCabaña(id)

**Responsabilidades:** Capturar cabaña y agregar modificaciones al sistema.

**Excepciones:**

**Referencias cruzadas:** Caso de uso: Modificar Cabaña

**Precondiciones:**

**Poscondiciones:**

Se creó una instancia de Cabaña.

Se cambiaron los atributos de Cabaña.

Se guardó la instancia de Cabaña en la base de datos.

### ***eliminarReserva(id):***

**Nombre:** eliminarReserva(id)

**Responsabilidades:** Eliminar una reserva de la base de datos.

**Excepciones:**

**Referencias cruzadas:** Caso de uso: Eliminar cabaña

**Precondiciones:**

**Poscondiciones:**

Se eliminó la reserva de la base de datos.

### ***eliminarCliente(id):***

**Nombre:** eliminarCliente(id)

**Responsabilidades:** Eliminar un cliente de la base de datos.

**Excepciones:**

**Referencias cruzadas:** Caso de uso: Eliminar cliente

**Precondiciones:**

**Poscondiciones:**

Se eliminó el cliente de la base de datos.

### ***enviarCorreo(id):***

**Nombre:** enviarCorreo(id)

**Responsabilidades:** Enviar correo al email de un cliente.

**Excepciones:** El correo no es válido.

**Referencias cruzadas:** Caso de uso: Enviar correo

**Precondiciones:**

**Poscondiciones:**

Se envió un correo al email del cliente.

### ***responderConsulta(respuesta):***

**Nombre:** responderConsulta(respuesta)

**Responsabilidades:** Enviar respuesta a la red social del cliente.

**Excepciones:**

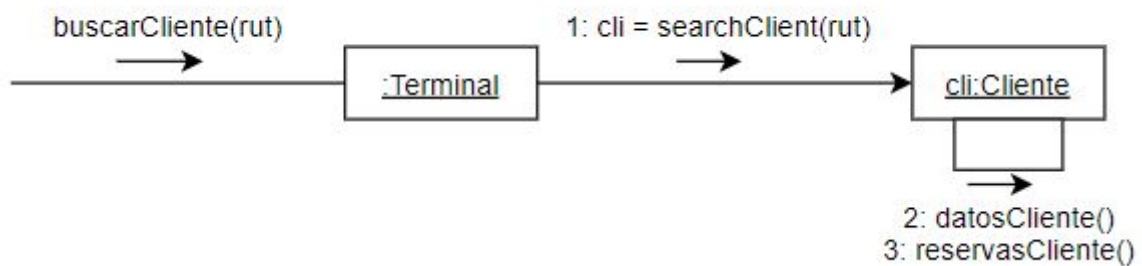
**Referencias cruzadas:** Caso de uso: Consultas redes sociales.

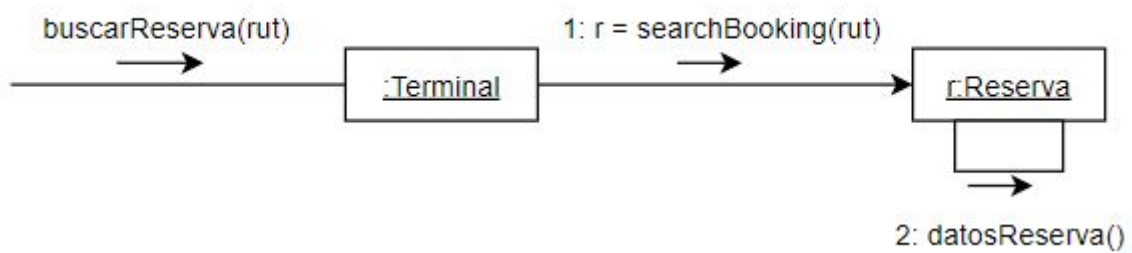
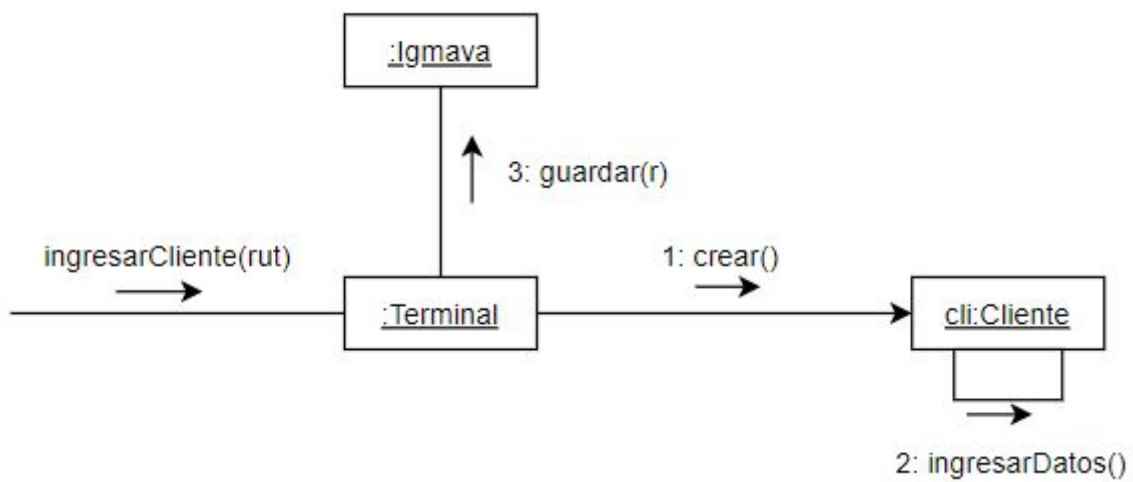
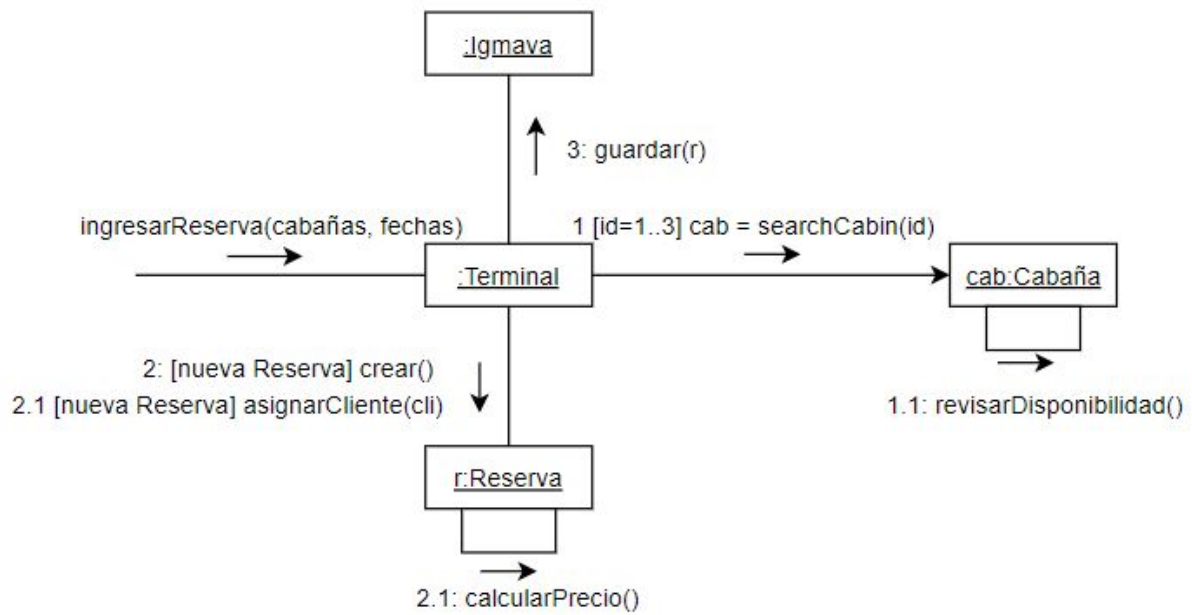
**Precondiciones:**

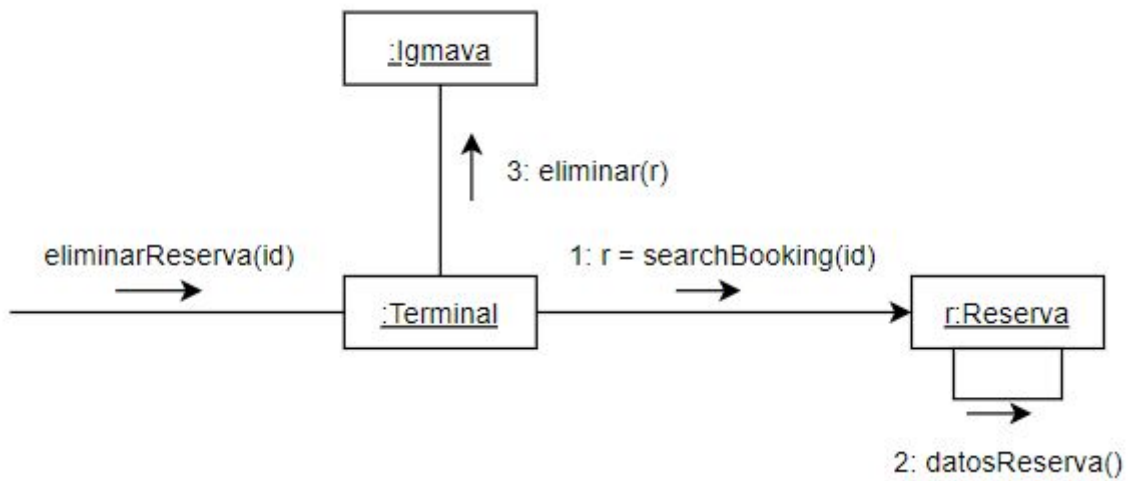
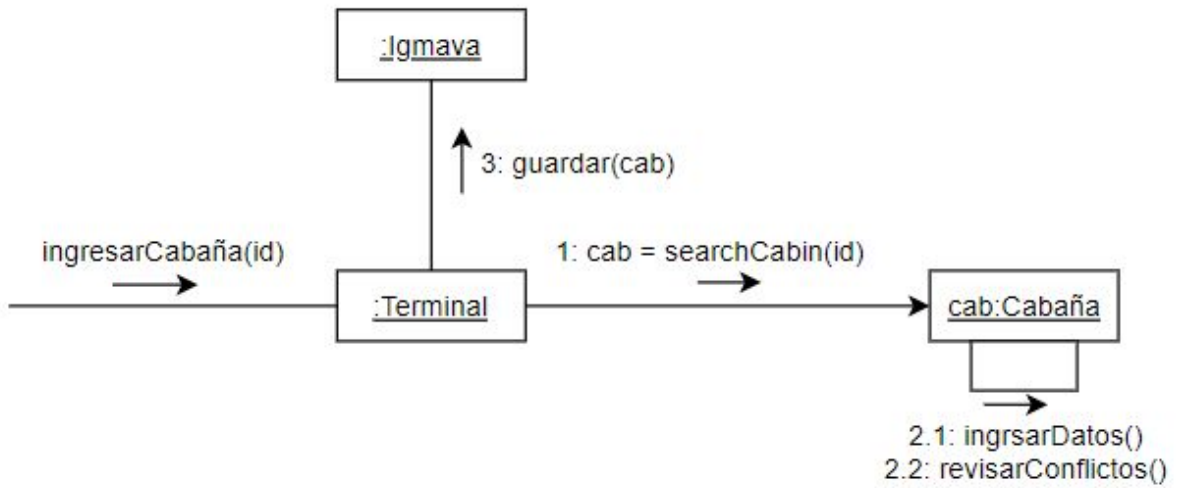
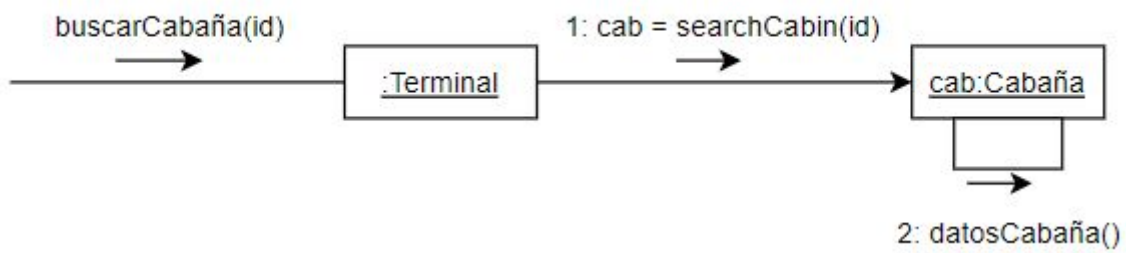
**Poscondiciones:**

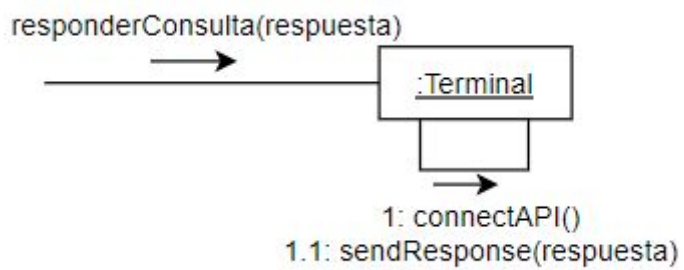
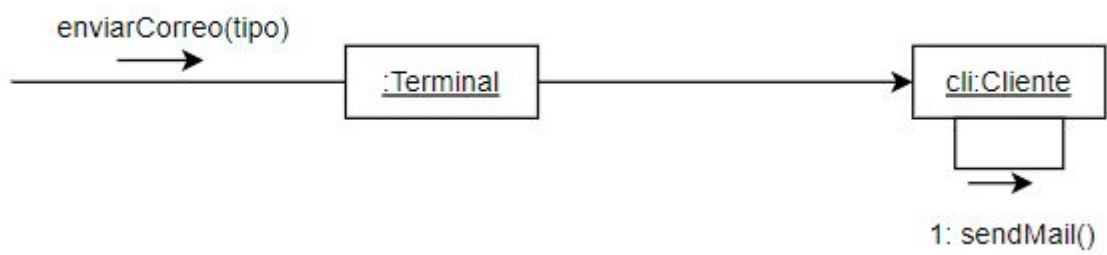
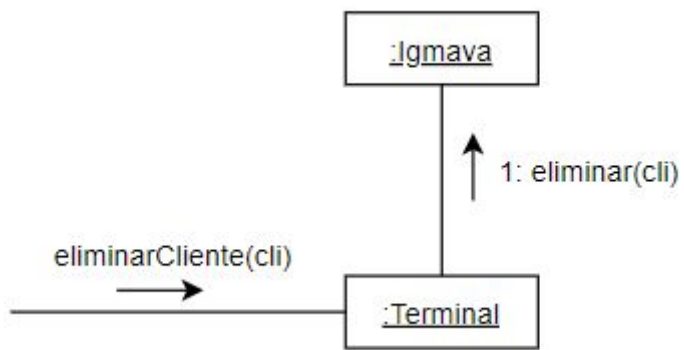
Se envió la respuesta a la red social del usuario.

## **Diagramas de colaboración**

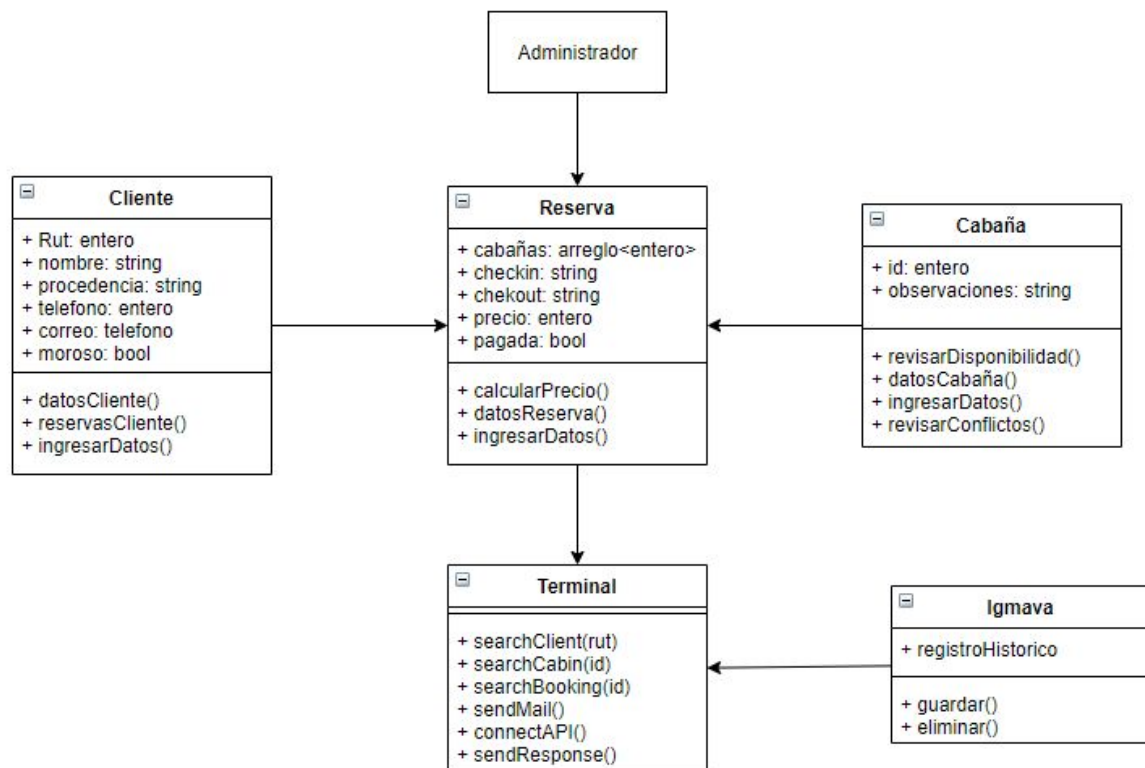








# Diagrama de clases



## Planificación

Para la planificación usamos el software Redmine, modelamos nuestro trabajo como una estructura por sprints, pues sentimos que es el que más representa nuestra metodología de trabajar en periodos de tiempo cortos y después tomar descansos del proyecto para poder dedicarle a otras asignaturas, nuestra carta Gantt generada en Redmine se ve de la siguiente manera:

