



Universidad Técnica Federico Santa María



Tarea 3 y 4

“Sistema de Gestión de Clientes VIP”

FUNDAMENTOS DE INGENIERÍA DE SOFTWARE

Rodrigo Fernández - 2673002-3
rfernand@inf.utfsm.cl

Cristián Maureira - 2673030-9
cmaureir@inf.utfsm.cl

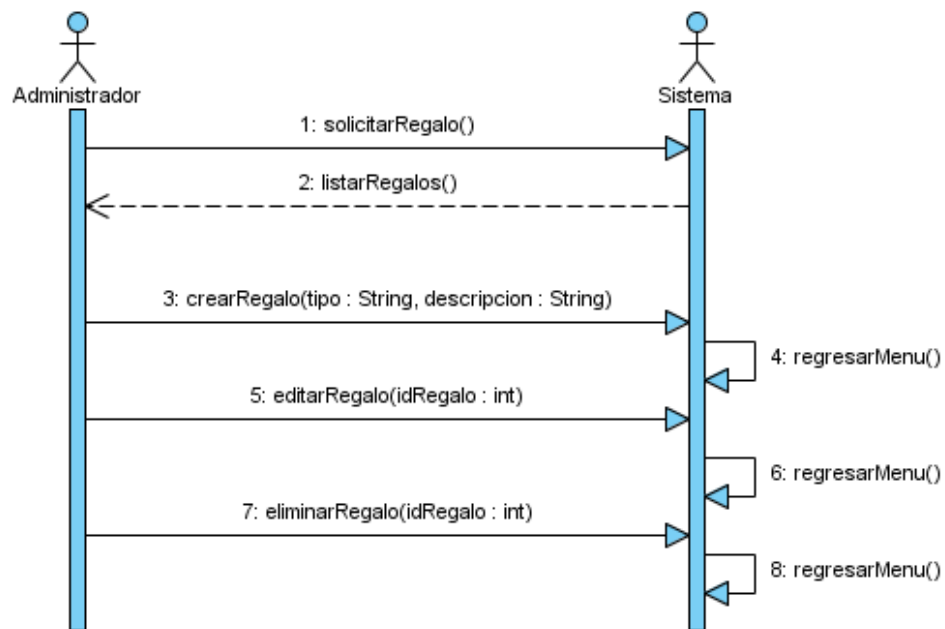
VALPARAÍSO, DICIEMBRE 2008.

1. Tarea 3

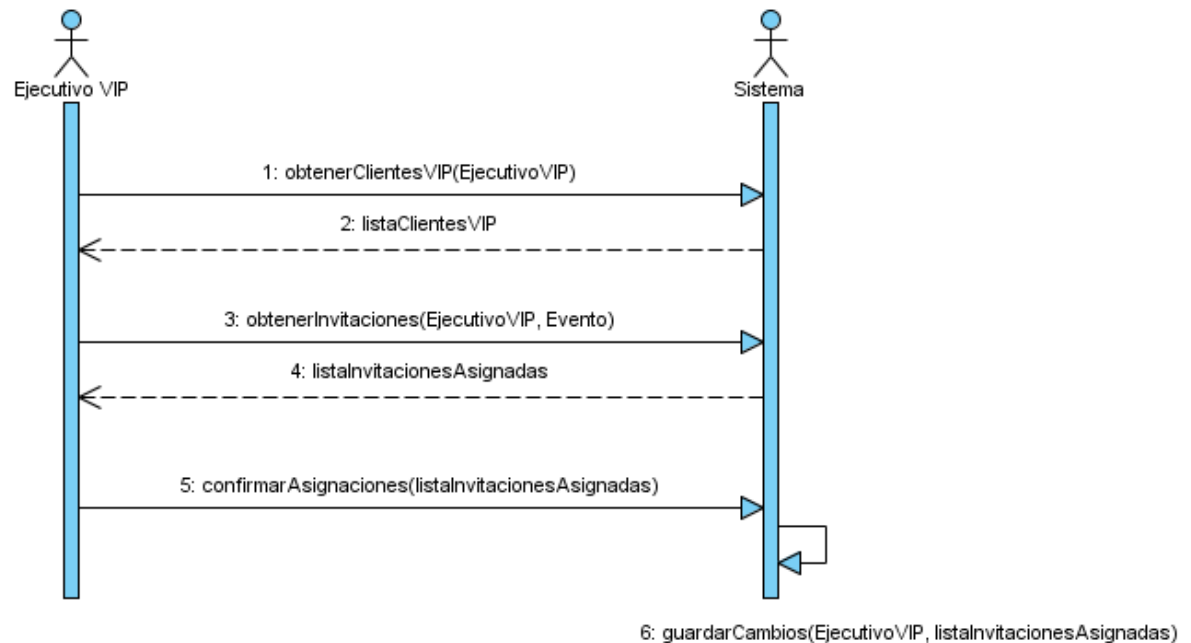
1.1. Diagramas de Secuencia del Sistema

1. DSS para Mantener un Regalo

sd mantenerRegalo /



2. DSS para Asignar una Entrada de un Evento a un Cliente VIP

sd Asignar Invitaciones a Clientes VIP**1.2. Contratos de las Operaciones Identificadas**

■ Contrato crearRegalo

Nombre:	crearRegalo
Parámetros:	string tipo , string descripcion
Responsabilidades:	Guardar en el sistema la definición de un nuevo Regalo (su tipo y descripción).
Tipo:	Sistema
Excepciones:	Ya existe el regalo a ingresado
Precondiciones:	El tipo de regalo debe ser único.
Postcondiciones:	Se crea una nueva instancia de Regalo, con idRegalo único.

■ Contrato editarRegalo

Nombre:	editarRegalo
Parámetros:	int idRegalo
Responsabilidades:	Despliega los atributos del Regalo correspondiente (tipo y descripción) para luego pasar a editar y guardar los cambios.
Tipo:	Sistema
Excepciones:	El Regalo a editar no existe
Precondiciones:	El idRegalo ingresado debe existir dentro de las instancias de Regalo.
Postcondiciones:	Se guardan los nuevos atributos en la instancia seleccionada.

- Contrato eliminarRegalo

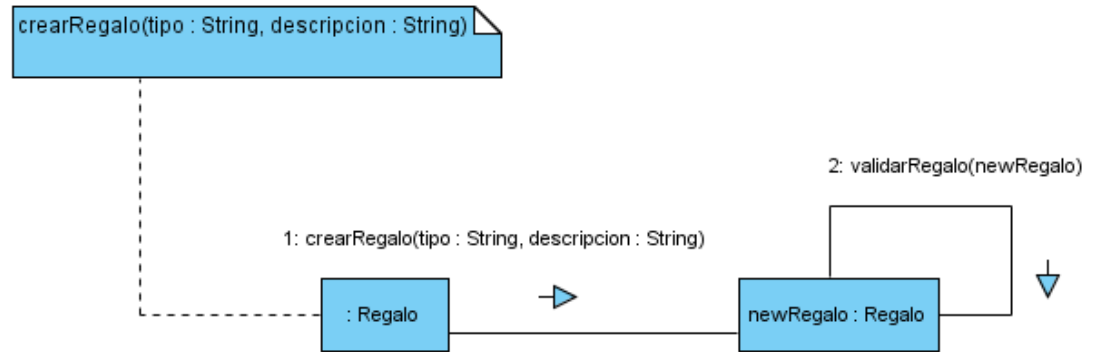
Nombre:	eliminarRegalo
Parámetros:	int idRegalo
Responsabilidades:	Elimina la instancia de Regalo que coincida con idRegalo
Tipo:	Sistema
Excepciones:	El Regalo a eliminar no existe
Precondiciones:	El idRegalo ingresado debe existir dentro de las instancias de Regalo.
Postcondiciones:	La instancia que coincida con idRegalo es eliminada.

- Contrato guardarCambios

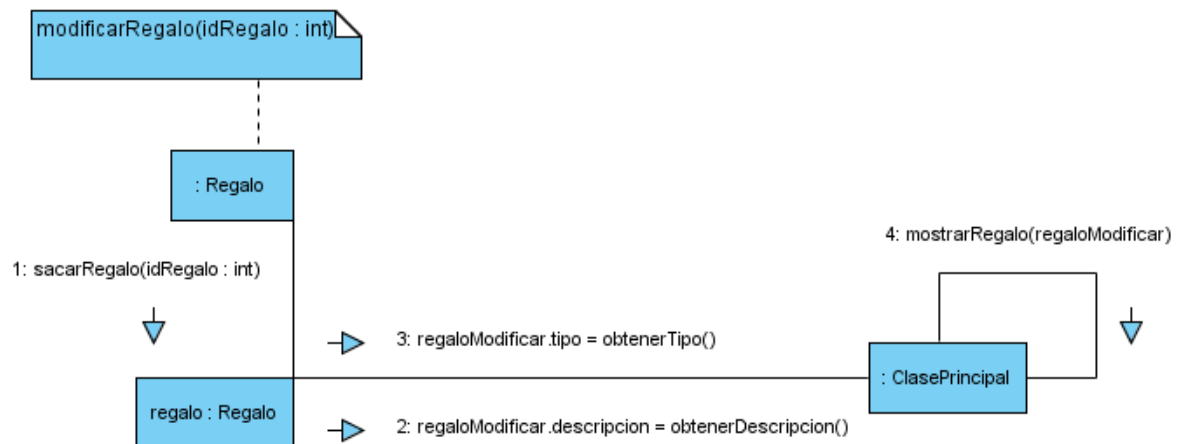
Nombre:	guardarCambios
Parámetros:	ejecutivoVIP, listaInvitacionesAsignadas
Responsabilidades:	Actualizar por cada Cliente VIP del ejecutivoVIP las invitaciones asignadas
Tipo:	Sistema
Excepciones:	El ejecutivoVIP no tenia ninguna invitación asignada
Precondiciones:	Se modificaron uno o más de los valores editables de las invitaciones asignadas de los usuarios para el evento
Postcondiciones:	Se eliminó todas las invitaciones actualmente asignadas a los clientesVIP del ejecutivoVIP. Se crearon invitaciones y se asignaron para el clienteVIP y el Evento segun hallan sido ingresadas anteriormente y se establecieron los estados respectivos a cada invitación.

1.3. Diagramas de Colaboracion de los Contratos

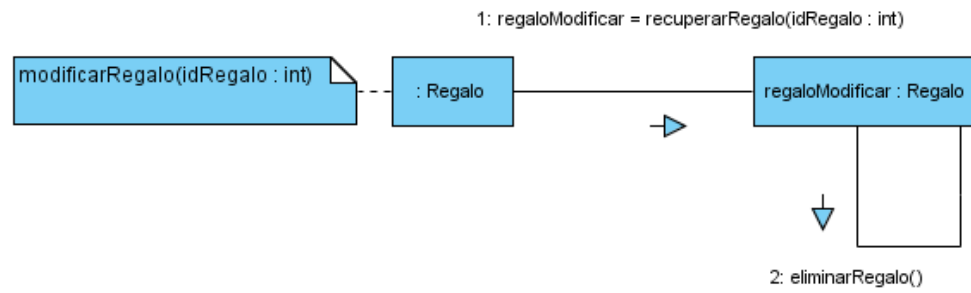
- crearRegalo

sd crearRegalo

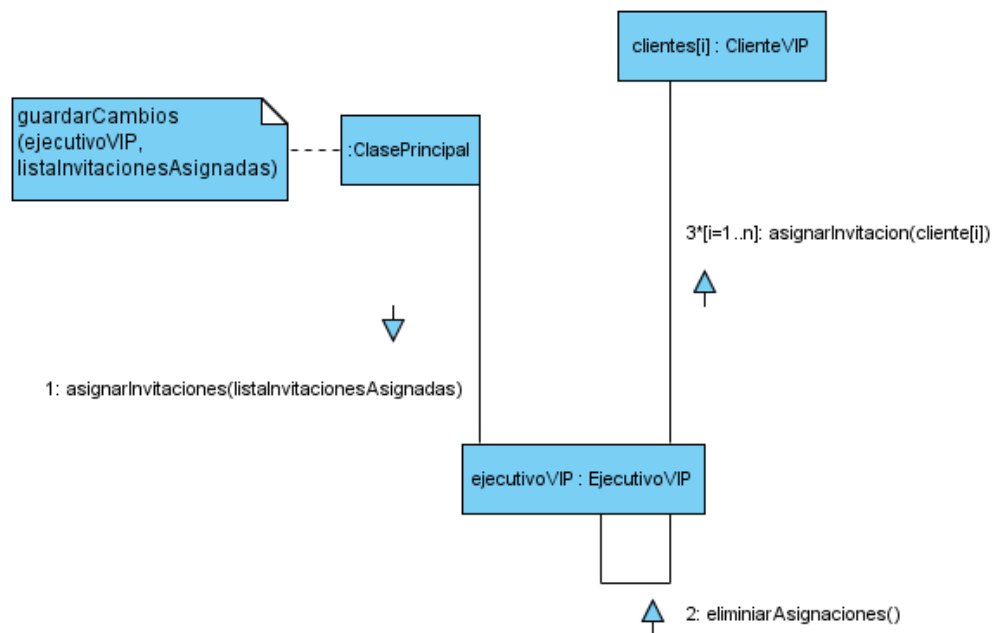
- `modificarRegalo`

sd modificarRegalo

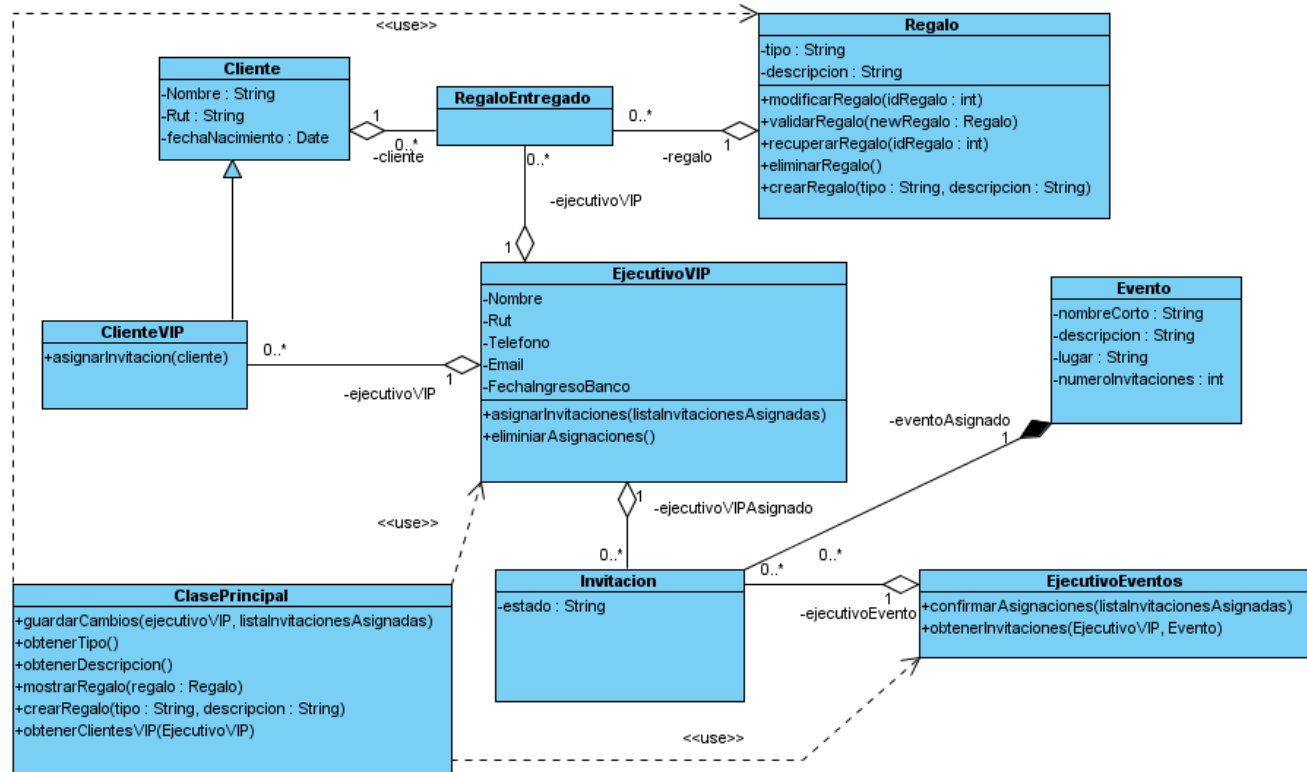
- `eliminarRegalo`

sd eliminarRegalo

- guardarCambios

sd guardarCambios

1.4. Diagramas de Clases



2. Supuestos Tarea 3

- Las llamadas iniciales de los diagramas de comunicación fueron representadas como notas dado que no logramos representarlas como una llamada no numerada (y no logramos comenzar una llamada del vacío, el programa no lo permitía).
- Se considero una clase 'clasePrincipal' que controla todas las operaciones del sistema general.
- Consideramos que seria bueno que el Administrador de Eventos sea identificable para poder reconocer a posterioridad que administrador de Eventos realizo los mejores eventos, o cometio errores al realizar algo.
- Se considero que el curso normal de los eventos, pueden haber diferentes Administadores del Sistema trabajando paralelamente, por lo que habia que validar los datos para evitar inconsistencias en la base de datos.
- Se adaptaron los diagramas según la pauta dada.

- Se considero que según el texto, el que mantenía los eventos era el Administrador de Eventos y no el Administrador del Sistema.

3. Tarea 4

3.1. Prototipo de las pantallas del sistema (no funcional)

Se adjunta un directorio con todas las vistas de usuarios.

Desde el *index.html* se pude navegar a todas las vistas(casos de uso), obviamente nada está funcional ya que para eso deberia apoyarme en algun lenguaje dinámico como *php*.

Siguiendo con lo mismo, no hay verificaciones, ni al momento de editar algo, que aparesca el valor anterior del campo.

3.2. Diagramas de Clases considerando los patrones de diseño de alto nivel

4. Supuestos Tarea 4

- Se considero una clase 'clasePrincipal' que controla todas las operaciones del sistema general.
- Consideramos que seria bueno que el Administrador de Eventos sea identificable para poder reconocer a posterioridad que administrador de Eventos realizo los mejores eventos, o cometio errores al realizar algo.
- Se considero que el curso normal de los eventos, pueden haber diferentes Administadores del Sistema trabajando paralelamente, por lo que habia que validar los datos para evitar inconsistencias en la base de datos.
- Se adaptó el diagrama de casos de uso segun la pauta dada en la tarea.
- Se considero que según el texto, el que mantenía los eventos era el Administrador de Eventos y no el Administrador del Sistema.