

DISEÑO ITERACIÓN 2

Alexander Moreno Borrego
Carlos Jesús Fernández Basso
Francisco Santolalla Quiñonero



21 DE ABRIL DE 2013
Universidad de Granada

Contenido

CONTROL DE VERSIONES.....	4
MODELADO DE REQUISITOS	5
MODELADO FUNCIONAL – DIAGRAMAS DE CASOS DE USO.....	5
Casos de uso: Gestión de actividades	7
Casos de uso: Gestión categorías	11
Casos de uso: Gestión de equipos	15
Casos de uso: Gestión de grupos	19
Casos de uso: Gestión de pagos	23
Casos de uso: Gestión de temporadas	27
Casos de uso: Gestión de instalaciones	31
IDENTIFICAR SUBSISTEMAS – DIAGRAMAS DE CASOS DE USO.....	35
Diagrama de CU: Gestión de Actividades.....	35
Diagrama de CU: Gestión de Grupos	36
Diagrama de CU: Gestión de Temporadas.....	37
Diagrama de CU: Gestión de Equipos	38
Diagrama de CU: Gestión de Instalaciones	39
Diagrama de CU: Gestión de Pagos	40
Diagrama de CU: Gestión de Categorías.....	41
REQUISITOS NO FUNCIONALES.....	42
OPERACIONES DEL SISTEMA – DIAGRAMAS DE SECUENCIA DE SISTEMA	43
Gestión de Actividades.....	43
Diagrama de secuencia:Insertar Actividad.....	43
Diagrama de secuencia:Consultar Actividad	44
Diagrama de secuencia:Eliminar Actividad.....	45
Diagrama de secuencia: ModificarActividad	46
Gestión de Categorías	47
Diagrama de secuencia: Insertar Categoría	47
Diagrama de secuencia: Consultar Categoría	48
Diagrama de secuencia: Eliminar Categoría.....	49
Diagrama de secuencia: Modificar Categoría	49
Gestión de Equipos	50
Diagrama de secuencia: Insertar Equipo	50
Diagrama de secuencia: Consultar Equipo.....	51
Diagrama de secuencia: Eliminar Equipo	52
Diagrama de secuencia: Modificar Equipo.....	53
Gestión de Grupos	54
Diagrama de secuencia: Insertar Grupo	54
Diagrama de secuencia: Consultar Grupo.....	55
Diagrama de secuencia: Eliminar Grupo	56
Diagrama de secuencia: Modificar Grupo.....	57
Gestión de Instalaciones	58
Diagrama de secuencia: Insertar Instalaciones	58
Diagrama de secuencia: Consultar Instalaciones	59
Diagrama de secuencia: Eliminar Instalaciones	60
Diagrama de secuencia: Modificar Instalaciones.....	61
Gestión de Pagos	62
Diagrama de secuencia: Insertar Pago.....	62
Diagrama de secuencia: Consultar Pago.....	63
Diagrama de secuencia: Eliminar Pago	64
Diagrama de secuencia: Modificar Pago	65
Gestión de Temporadas.....	66
Diagrama de secuencia: Insertar Temporada.....	66
Diagrama de secuencia: Consultar Temporada.....	67
Diagrama de secuencia: Eliminar Temporada	68
Diagrama de secuencia: Modificar Temporada	69

ANÁLISIS.....	70
IDENTIFICAR CLASES, ATRIBUTOS Y RELACIONES	70
Clase: Actividad.....	70
Clase: Categoría.....	70
Clase: Equipo	71
Clase: Grupo Entrenamiento.....	71
Clase: Pago Temporada (clase de asociación entre las clases Temporada y Alumno).....	71
Clase: Pago Actividad (clase de asociación entre Alumno y Actividad)	72
Clase: Cuota Precio.....	72
Clase: Temporada.....	72
Clase: Instalación	73
Clase: Horario (clase de asociación entre Instalación y GrupoEntrenamiento)	73
MODELADO ESTÁTICO – DIAGRAMA DE CLASES.....	74
MODELADO DEL COMPORTAMIENTO EXTERNO – CONTRATOS	75
Contratos de los casos de uso de la Gestión de Actividades	75
Contratos de Caso de uso: Insertar Actividad.....	75
Contratos de Caso de uso: Consultar Actividad	77
Contratos de Caso de uso: Eliminar Actividad.....	77
Contratos de Caso de uso: Modificar Actividad	78
Contratos de los casos de uso de la Gestión de Categorías	79
Contratos de Caso de uso: Insertar Categoría	79
Contratos de Caso de uso: Consultar Categoría	79
Contratos de Caso de uso: Eliminar Categoría.....	80
Contratos de Caso de uso: Modificar Categoría	80
Contratos de los casos de uso de la Gestión de Equipos	81
Contratos de Caso de uso: Insertar Equipo.....	81
Contratos de Caso de uso: Consultar Equipo.....	82
Contratos de Caso de uso: Eliminar Equipo	82
Contratos de Caso de uso: Modificar Equipo.....	83
Contratos de los casos de uso de la Gestión de Grupos.....	84
Contratos de Caso de uso: Insertar Grupo	84
Contratos de Caso de uso: Consultar Grupo.....	85
Contratos de Caso de uso: Eliminar Grupo	85
Contratos de Caso de uso: Modificar Grupo	86
Contratos de los casos de uso de la Gestión de Instalaciones	87
Contratos de Caso de uso: Insertar Instalación	87
Contratos de Caso de uso: Consultar Instalación.....	87
Contratos de Caso de uso: Eliminar Instalación	88
Contratos de Caso de uso: Modificar Instalación.....	88
Contratos de los casos de uso de la Gestión de Pagos	89
Contratos de Caso de uso: Insertar Categoría	89
Contratos de Caso de uso: Consultar Categoría	90
Contratos de Caso de uso: Eliminar Pago.....	91
Contratos de Caso de uso: Modificar Pago	92
Contratos de los casos de uso de la Gestión de Temporadas	93
Contratos de Caso de uso: Insertar Temporada.....	93
Contratos de Caso de uso: Consultar Temporada.....	93
Contratos de Caso de uso: Eliminar Temporada	94
Contratos de Caso de uso: Modificar Temporada	94
DISEÑO.....	95
DESCOMPOSICIÓN DEL SISTEMA EN SUBSISTEMAS DE DISEÑO PARA OBTENER LA ARQUITECTURA DEL SISTEMA.....	95
Establecer la arquitectura del sistema.	95
Subsistemas funcionales.....	95
Subsistema Gestión de Actividades	96
Subsistema Gestión de Categorías	96
Subsistema Gestión de Equipos.....	97
Subsistema Gestión de Grupos	97

Subsistema Gestión de Instalaciones	98
Subsistema Gestión de Temporadas.....	98
Subsistema Gestión de Pagos.....	99
Determinación de la arquitectura.	100
Diagrama arquitectura software de tres capas cerradas	100
DIAGRAMAS DE SECUENCIA DE DISEÑO	101
Diagrama de secuencia: Consultar Actividad	101
Diagrama de secuencia: Eliminar Actividad.....	101
Diagrama de secuencia: Insertar Actividad.....	102
Diagrama de secuencia: Modificar Actividad	103
Diagrama de secuencia: Consultar Categorías	104
Diagrama de secuencia: Eliminar Categorías.....	104
Diagrama de secuencia: Insertar Categorías.....	104
Diagrama de secuencia: Modificar Categorías	105
Diagrama de secuencia: Consultar Instalaciones.....	105
Diagrama de secuencia: Eliminar Instalaciones	105
Diagrama de secuencia: Insertar Instalaciones	106
Diagrama de secuencia: Modificar Instalaciones.....	106
Diagrama de secuencia: Consultar Pagos.....	106
Diagrama de secuencia: Eliminar Pagos	107
Diagrama de secuencia: Insertar Pagos.....	107
Diagrama de secuencia: Modificar Pagos.....	107
Diagrama de secuencia: Consultar Equipos	110
Diagrama de secuencia: Eliminar Equipos	110
Diagrama de secuencia: Insertar Equipos	111
Diagrama de secuencia: Modificar Equipos.....	112
Diagrama de secuencia: Consultar Grupos	112
Diagrama de secuencia: Eliminar Grupos.....	113
Diagrama de secuencia: Insertar Grupos	114
Diagrama de secuencia: Modificar Grupos.....	115
Diagrama de secuencia: Consultar Pagos.....	116
Diagrama de secuencia: Eliminar Pagos	116
Diagrama de secuencia: Insertar Pagos.....	116
Diagrama de secuencia: Modificar Pagos.....	117
PASO DE CLASES A MODELO RELACIONAL.....	117
MODELAR DIAGRAMA DE COMPONENTES	118
Diagrama de Componentes	119
ENCAJAR EL DIAGRAMA DE CLASES (OBTENIDO ANTERIORMENTE) EN LA ARQUITECTURA OBTENIDA EN EL APARTADO ANTERIOR.....	120
DIAGRAMA DE DESPLIEGUE DE DISEÑO.....	120
Diagrama de despliegue	121

Control de Versiones

Fecha	Versión	Descripción
10/3/2013	2.1	Añadidos casos de uso (instalaciones, equipos, grupos pagos, temporadas, actividades), diagramas de casos de uso y diagramas de secuencia
17/4/2013	2.2	Realizadas las modificaciones y correcciones indicadas por el equipo de planificación y el profesor
19/4/2012	2.3	Realizada la tarea de análisis de la fase de diseño
19/4/2012	2.4	Realizada la tarea de diseño de la fase de diseño

Modelado de requisitos

Modelado funcional – Diagramas de casos de uso

En esta etapa del modelado de requisitos se captura el propósito general del sistema:

- Se analiza qué debe hacer el sistema.
- Se obtiene una visión contextualizada del sistema.
- Identifica y delimita el sistema.
- Se determinan características, cualidades y restricciones que debe satisfacer el sistema.

Los requisitos funcionales que se han obtenido en el sistema son los siguientes:

- Un entrenador puede modificar los datos de sus alumnos relativos al grupo (comentarios, observaciones).
- El administrador es el encargado de gestionar las fichas de los alumnos, así como algunas actividades esporádicas.
- La notificación de los pagos mensuales de los alumnos se recibirá mediante un tipo de archivo procesable por nuestro sistema (sql, xml, etc).
- Hay que llevar un inventario de las equipaciones entregadas o no por cada temporada.
- Cada grupo de entrenamiento se compone de entre 15-20 alumnos y se distribuyen entre dos días de la semana a especificar.
- No puede haber dos grupos de entrenamiento a la misma hora, al mismo día y en la misma pista de la instalación.
- Un alumno solo puede jugar en el equipo al que está inscrito y puede competir en dos equipos, uno de su categoría y otro de una superior.
- La edad mínima de los alumnos tiene que ser 5 años, siendo la máxima 18 años.
- Los alumnos menores de 9 años, no tienen por qué pertenecer a un equipo.
- Los alumnos mayores de 9 años tienen que competir en al menos un equipo.
- La categoría benjamín y alevín solo pueden competir y entrenar en las pistas de mini-basket de las instalaciones.
- Se hará distinción entre 2 tipos de usuario para el acceso: administrador y entrenador. Cada uno de ellos podrá visualizar una parte distinta de la aplicación dependiendo de los privilegios que tenga.
- El administrador será el encargado de introducir los datos de los entrenadores en el sistema y proporcionar los usuarios y claves asociadas a cada entrenador.
- El administrador podrá modificar las contraseñas del sistema una vez creadas las cuentas en el mismo.
- Hay 5 categorías posibles: benjamín, alevín, infantil, cadete y junior.

En nuestro sistema hemos identificado los siguientes casos de uso:

Administrador	Entrenador
Gestión de actividades: <ul style="list-style-type: none"> ✓ Insertar actividad ✓ Eliminar actividad ✓ Modificar actividad ✓ Consultar actividad Gestión de categorías: <ul style="list-style-type: none"> ✓ Insertar categoría ✓ Eliminar categoría ✓ Modificar categoría ✓ Consultar categoría Gestión de equipos: <ul style="list-style-type: none"> ✓ Insertar equipo ✓ Dar de baja equipo ✓ Modificar equipo ✓ Consultar equipo Gestión de grupos: <ul style="list-style-type: none"> ✓ Insertar grupos ✓ Dar de baja grupos ✓ Modificar grupos ✓ Consultar grupos Gestión de pagos: <ul style="list-style-type: none"> ✓ Insertar pagos ✓ Consultar pagos ✓ Modificar pagos ✓ Eliminar pago Gestión de temporadas: <ul style="list-style-type: none"> ✓ Insertar temporada ✓ Modificar temporada ✓ Consultar temporada ✓ Eliminar temporada Gestión de instalaciones: <ul style="list-style-type: none"> ✓ Insertar instalación ✓ Modificar instalación ✓ Consultar instalaciones ✓ Eliminar instalación 	Gestión de actividades <ul style="list-style-type: none"> ✓ Consultar actividad Gestión de categorías <ul style="list-style-type: none"> ✓ Consultar categoría Gestión de equipo <ul style="list-style-type: none"> ✓ Modificar equipo ✓ Consultar equipo Gestión de grupos <ul style="list-style-type: none"> ✓ Modificar grupos ✓ Consultar grupos Gestión de temporadas <ul style="list-style-type: none"> ✓ Consultar temporada Gestión de instalaciones <ul style="list-style-type: none"> ✓ Consultar instalaciones

El **actor administrador** será el encargado de realizar la mayor parte de operaciones del sistema para las altas, bajas y modificaciones de los datos del mismo.

El **actor entrenador** sólo podrá acceder a ciertos contenidos del sistema para consultar información o para en ciertos casos, modificar información relativa a los alumnos en el campo observaciones del mismo.

A continuación se detallan la descripción de los casos de uso del sistema.

Casos de uso: Gestión de actividades

Nombre del CU	Insertar actividad
Resumen	Se permite al administrador introducir en el sistema información relacionada con una nueva actividad.
Dependencias	-
Actores	Administrador
Precondiciones	La información de la actividad no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	La nueva actividad se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de dar de alta una actividad. 2. El sistema muestra un formulario con los campos a rellenar de la nueva actividad al administrador. <ol style="list-style-type: none"> 2.1. Opcionalmente se insertan los alumnos que van a disfrutar de la actividad. 2.2. Opcionalmente se asignan las instalaciones que se crean necesarias para llevar a cabo la actividad. 3. El administrador rellena el formulario con todos los campos necesarios y guarda la actividad.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la actividad. 3b. La nueva actividad ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Modificar actividad
Resumen	Se permite al administrador modificar los datos de una actividad del sistema.
Dependencias	-
Actores	Administrador.
Precondiciones	La actividad a modificar debe existir previamente en el sistema y se sabrá qué actividad se va a modificar. El administrador debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de modificar actividad. 2. El sistema muestra un formulario con los datos actuales de la actividad en cuestión. <ol style="list-style-type: none"> 2.1. Opcionalmente se insertan o eliminan los alumnos que van a disfrutar de la actividad. 2.2. Opcionalmente se reasignan las instalaciones que se crean necesarias para llevar a cabo la actividad. 3. El administrador modifica los campos que desee y guarda la actividad.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la actividad. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar actividad
Resumen	Se permite al usuario consultar del sistema información relacionada con una actividad existente.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	Debe de existir la actividad que se quiere consultar para poder llevar a cabo la operación. El usuario debe estar identificado en el sistema.
Postcondición	El sistema muestra al usuario las actividades consultadas.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de consultar actividad. 2. El sistema muestra los campos por los que se pueden buscar actividades. 3. El usuario rellena uno o varios de los campos para buscarlas y ejecuta la búsqueda. 4. El sistema muestra una lista con las actividades que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ninguna actividad, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar actividad
Resumen	Se permite al administrador eliminar del sistema información relacionada con una actividad existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir la actividad que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué actividad se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	La actividad eliminada ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar actividad. 2. El sistema muestra un mensaje de confirmación de eliminación de la correspondiente actividad. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema la actividad.
Cursos alternativos	3a. El administrador rechaza la eliminación de la actividad.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Casos de uso: Gestión categorías

Nombre del CU	Insertar categoría
Resumen	Se permite al administrador introducir en el sistema información relacionada con una nueva categoría.
Dependencias	-
Actores	Administrador
Precondiciones	La información de la categoría no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	La nueva categoría se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de dar de alta una categoría. 2. El sistema muestra un formulario con los campos a rellenar de la nueva categoría al administrador. 3. El administrador rellena el formulario con todos los campos necesarios y guarda la categoría.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la categoría. 3b. La nueva categoría ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Modificar categoría
Resumen	Se permite al administrador modificar los datos de una categoría del sistema.
Dependencias	-
Actores	Administrador.
Precondiciones	La categoría a modificar debe existir previamente en el sistema y se sabrá qué categoría se va a modificar. El administrador debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de modificar categoría. 2. El sistema muestra un formulario con los datos actuales de la categoría en cuestión. 3. El administrador modifica los campos que desee y guarda la categoría.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la categoría. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar categoría
Resumen	Se permite al usuario consultar del sistema información relacionada con una categoría existente.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	Debe de existir la categoría que se quiere consultar para poder llevar a cabo la operación. El usuario debe estar identificado en el sistema.
Postcondición	El sistema muestra al usuario las categorías consultadas.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de consultar categoría. 2. El sistema muestra los campos por los que se pueden buscar categorías. 3. El usuario rellena uno o varios de los campos para buscarlas y ejecuta la búsqueda. 4. El sistema muestra una lista con las categorías que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ninguna categoría, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar categoría
Resumen	Se permite al administrador eliminar del sistema información relacionada con una categoría existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir la categoría que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué categoría se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	La categoría eliminada ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar categoría. 2. El sistema muestra un mensaje de confirmación de eliminación de la correspondiente categoría. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema la categoría.
Cursos alternativos	3a. El administrador rechaza la eliminación de la categoría.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Casos de uso: Gestión de equipos

Nombre del CU	Insertar equipo
Resumen	Se permite al administrador introducir en el sistema información relacionada con un nuevo equipo.
Dependencias	-
Actores	Administrador
Precondiciones	La información del equipo no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	El nuevo equipo se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de dar de alta un equipo. 2. El sistema muestra un formulario con los campos a rellenar del nuevo equipo al administrador. <ol style="list-style-type: none"> 2.1. Opcionalmente se insertan los alumnos que van a formar parte del equipo. 3. El administrador rellena el formulario con todos los campos necesarios y guarda el equipo.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar el equipo. 3b. El nuevo equipo ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve al paso 2.
Observaciones	El equipo se asignará a una temporada en el paso 2.
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Modificar equipo
Resumen	Se permite al usuario modificar los datos de un equipo del sistema.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	El equipo a modificar debe existir previamente en el sistema y se sabrá qué equipo se va a modificar. El usuario debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de modificar equipo. 2. El sistema muestra un formulario con los datos actuales del equipo en cuestión. <ol style="list-style-type: none"> 2.1. Opcionalmente se insertan o eliminan los alumnos que van a formar parte del equipo. 3. El usuario modifica los campos que desee y guarda el equipo.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El usuario cierra la ventana sin guardar el equipo. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar equipo
Resumen	Se permite al usuario consultar del sistema información relacionada con un equipo existente.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	Debe de existir un equipo que se quiere consultar para poder llevar a cabo la operación. El usuario debe estar identificado en el sistema.
Postcondición	El sistema muestra al usuario los equipos consultados.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de consultar equipo. 2. El sistema muestra los campos por los que se pueden buscar equipos. 3. El usuario rellena uno o varios de los campos para buscarlos y ejecuta la búsqueda. 4. El sistema muestra una lista con los equipos que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ningún equipo, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar equipo
Resumen	Se permite al administrador eliminar del sistema información relacionada con un equipo existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir un equipo que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué equipo se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	Equipo eliminado ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar equipo. 2. El sistema muestra un mensaje de confirmación de eliminación del correspondiente equipo. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema el equipo.
Cursos alternativos	4a. El administrador rechaza la eliminación del equipo.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Casos de uso: Gestión de grupos

Nombre del CU	Insertar grupo
Resumen	Se permite al administrador introducir en el sistema información relacionada con un nuevo grupo.
Dependencias	-
Actores	Administrador
Precondiciones	La información del grupo no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	El nuevo grupo se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de dar de alta un grupo. 2. El sistema muestra un formulario con los campos a rellenar del nuevogrupo al administrador. <ol style="list-style-type: none"> 2.1. Opcionalmente se insertan los alumnos que van a formar parte del grupo. 3. El administrador rellena el formulario con todos los campos necesarios y guarda el grupo.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar el grupo. 3b. El nuevogrupo ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	El grupo se asignará a una temporada en el paso 2.
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Modificar grupo
Resumen	Se permite al usuario modificar los datos de un grupo del sistema.
Dependencias	-
Actores	Administrador, entrenador
Precondiciones	El grupo a modificar debe existir previamente en el sistema y se sabrá qué grupo se va a modificar. El usuario debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de modificar grupo. 2. El sistema muestra un formulario con los datos actuales del grupo en cuestión. <ol style="list-style-type: none"> 2.2. Opcionalmente se insertan o eliminan los alumnos que van a formar parte del grupo. 3. El usuario modifica los campos que desee y guarda el grupo.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El usuario cierra la ventana sin guardar el grupo. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar grupo
Resumen	Se permite al usuario consultar del sistema información relacionada con un grupo existente.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	Debe de existir el grupo que se quiere consultar para poder llevar a cabo la operación. El usuario debe estar identificado en el sistema.
Postcondición	El sistema muestra al usuario los grupos consultados.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de consultar grupo. 2. El sistema muestra los campos por los que se pueden buscar grupos. 3. El usuario rellena uno o varios de los campos para buscarlos y ejecuta la búsqueda. 4. El sistema muestra una lista con los grupos que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ningún grupo, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar grupo
Resumen	Se permite al administrador eliminar del sistema información relacionada con un grupo existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir el grupo que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué grupo se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	El grupo eliminado ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar grupo. 2. El sistema muestra un mensaje de confirmación de eliminación del correspondiente grupo. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema el grupo.
Cursos alternativos	3a. El administrador rechaza la eliminación del grupo.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Casos de uso: Gestión de pagos

Nombre del CU	Insertar pago
Resumen	Se permite al administrador introducir en el sistema información relacionada con un nuevo pago.
Dependencias	-
Actores	Administrador
Precondiciones	La información del pago no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	El nuevo pago se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de insertar un pago. También elegirá si desea insertar un pago de una temporada o de una actividad. 2. El sistema muestra un formulario con los campos a rellenar del nuevopago al administrador. 3. El administrador rellena el formulario con todos los campos necesarios y guarda el pago.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar el pago. 3b. El nuevopago ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Modificar pago
Resumen	Se permite al administrador modificar los datos de un pago del sistema.
Dependencias	-
Actores	Administrador.
Precondiciones	El pago a modificar debe existir previamente en el sistema y se sabrá qué pago se va a modificar. El administrador debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de modificar pago. También elegirá si desea modificar un pago de una temporada o de una actividad. 2. El sistema muestra un formulario con los datos actuales del pago en cuestión. 3. El administrador modifica los campos que desee y guarda el pago.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar el pago. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales	no Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar pago
Resumen	Se permite al administrador consultar del sistema información relacionada con un pago existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir el pago que se quiere consultar para poder llevar a cabo la operación. El administrador debe estar identificado en el sistema.
Postcondición	El sistema muestra al administrador los pagos consultados.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de consultar pago. También elegirá si desea consultar un pago de una temporada o de una actividad. 2. El sistema muestra los campos por los que se pueden buscar pagos. 3. El administrador rellena uno o varios de los campos para buscarlos y ejecuta la búsqueda. 4. El sistema muestra una lista con los pagos que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ningún pago, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar pago
Resumen	Se permite al administrador eliminar del sistema información relacionada con un pago existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir el pago que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué pago se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	El grupo eliminado ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar pago. También elegirá si desea eliminar un pago de una temporada o de una actividad. 2. El sistema muestra un mensaje de confirmación de eliminación del correspondiente pago. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema el pago.
Cursos alternativos	3a. El administrador rechaza la eliminación del pago.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Casos de uso: Gestión de temporadas

Nombre del CU	Insertar temporada
Resumen	Se permite al administrador introducir en el sistema información relacionada con una nuevatemporada.
Dependencias	-
Actores	Administrador
Precondiciones	La información delatemporada no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	La nuevatemporada se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de insertar unatemporada. 2. El sistema muestra un formulario con los campos a rellenar dela nuevatemporada al administrador. 3. El administrador rellena el formulario con todos los campos necesarios y guarda latemporada.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar latemporada. 3b. La nuevatemporada ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales	no Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortara la ejecución del proceso.

Nombre del CU	Modificar temporada
Resumen	Se permite al administrador modificar los datos de una temporada del sistema.
Dependencias	-
Actores	Administrador.
Precondiciones	La temporada a modificar debe existir previamente en el sistema y se sabrá qué temporada se va a modificar. El administrador debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de modificar temporada. 2. El sistema muestra un formulario con los datos actuales de la temporada en cuestión. 3. El administrador modifica los campos que desee y guarda la temporada.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la temporada. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar temporada
Resumen	Se permite al usuario consultar del sistema información relacionada con una temporada existente.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	Debe de existir la temporada que se quiere consultar para poder llevar a cabo la operación. El usuario debe estar identificado en el sistema.
Postcondición	El sistema muestra al usuario las temporadas consultadas.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de consultar temporada. 2. El sistema muestra los campos por los que se pueden buscar temporadas. 3. El usuario rellena uno o varios de los campos para buscarlos y ejecuta la búsqueda. 4. El sistema muestra una lista con las temporadas que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ninguna temporada, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar temporada
Resumen	Se permite al administrador eliminar del sistema información relacionada con una temporada existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir la temporada que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué temporada se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	La temporada eliminada ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar temporada. 2. El sistema muestra un mensaje de confirmación de eliminación de la correspondiente temporada. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema la temporada.
Cursos alternativos	3a. El administrador rechaza la eliminación de la temporada.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Casos de uso: Gestión de instalaciones

Nombre del CU	Insertar instalación
Resumen	Se permite al administrador introducir en el sistema información relacionada con una nueva instalación.
Dependencias	-
Actores	Administrador
Precondiciones	La información de la instalación no debe existir previamente en el sistema. El administrador debe estar identificado en el sistema.
Postcondición	La nueva instalación se ha creado con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de dar de alta una instalación. 2. El sistema muestra un formulario con los campos a rellenar de la nueva instalación al administrador. 3. El administrador rellena el formulario con todos los campos necesarios y guarda la instalación.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la instalación. 3b. La nueva instalación ya existe en el sistema. El sistema informa del error y vuelve al paso 2. 3c. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Modificar instalación
Resumen	Se permite al administrador modificar los datos de una instalación del sistema.
Dependencias	-
Actores	Administrador.
Precondiciones	La instalación a modificar debe existir previamente en el sistema y se sabrá qué instalación se va a modificar. El administrador debe estar identificado en el sistema.
Postcondición	Los cambios realizados se almacenan con éxito.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de modificar instalación. 2. El sistema muestra un formulario con los datos actuales de la instalación en cuestión. 3. El administrador modifica los campos que desee y guarda la instalación.
Cursos alternativos	<ol style="list-style-type: none"> 3a. El administrador cierra la ventana sin guardar la instalación. 3b. Falta algún dato por rellenar. El sistema informa del error y vuelve el paso 2.
Observaciones	-
Requisitos funcionales	no Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Consultar instalación
Resumen	Se permite al usuario consultar del sistema información relacionada con una instalación existente.
Dependencias	-
Actores	Administrador, Entrenador
Precondiciones	Debe de existir la instalación que se quiere consultar para poder llevar a cabo la operación. El usuario debe estar identificado en el sistema.
Postcondición	El sistema muestra al usuario las instalaciones consultadas.
Curso normal	<ol style="list-style-type: none"> 1. El usuario selecciona la opción de consultar instalación. 2. El sistema muestra los campos por los que se pueden buscar instalaciones. 3. El usuario rellena uno o varios de los campos para buscarlas y ejecuta la búsqueda. 4. El sistema muestra una lista con las instalaciones que coinciden con los criterios de búsqueda.
Cursos alternativos	4a. El sistema no encuentra ninguna instalación, lanza un error y vuelve al paso 2.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Nombre del CU	Eliminar instalación
Resumen	Se permite al administrador eliminar del sistema información relacionada con una instalación existente.
Dependencias	-
Actores	Administrador
Precondiciones	Debe de existir la instalación que se quiere eliminar para poder llevar a cabo la operación y se sabrá qué instalación se va a eliminar. El administrador debe estar identificado en el sistema.
Postcondición	La instalación eliminada ya no formará parte del sistema.
Curso normal	<ol style="list-style-type: none"> 1. El administrador selecciona la opción de eliminar instalación. 2. El sistema muestra un mensaje de confirmación de eliminación de la correspondiente instalación. 3. El administrador confirma la decisión. 4. El sistema elimina del sistema la instalación.
Cursos alternativos	3a. El administrador rechaza la eliminación de la instalación.
Observaciones	-
Requisitos funcionales no	Cuando el sistema detecte una anomalía mostrará al usuario el mensaje de error pertinente y abortará la ejecución del proceso.

Identificar subsistemas – Diagramas de casos de uso

A continuación se detallan los distintos diagramas de casos de uso de los que se compone el sistema analizado.

Diagrama de CU: Gestión de Actividades

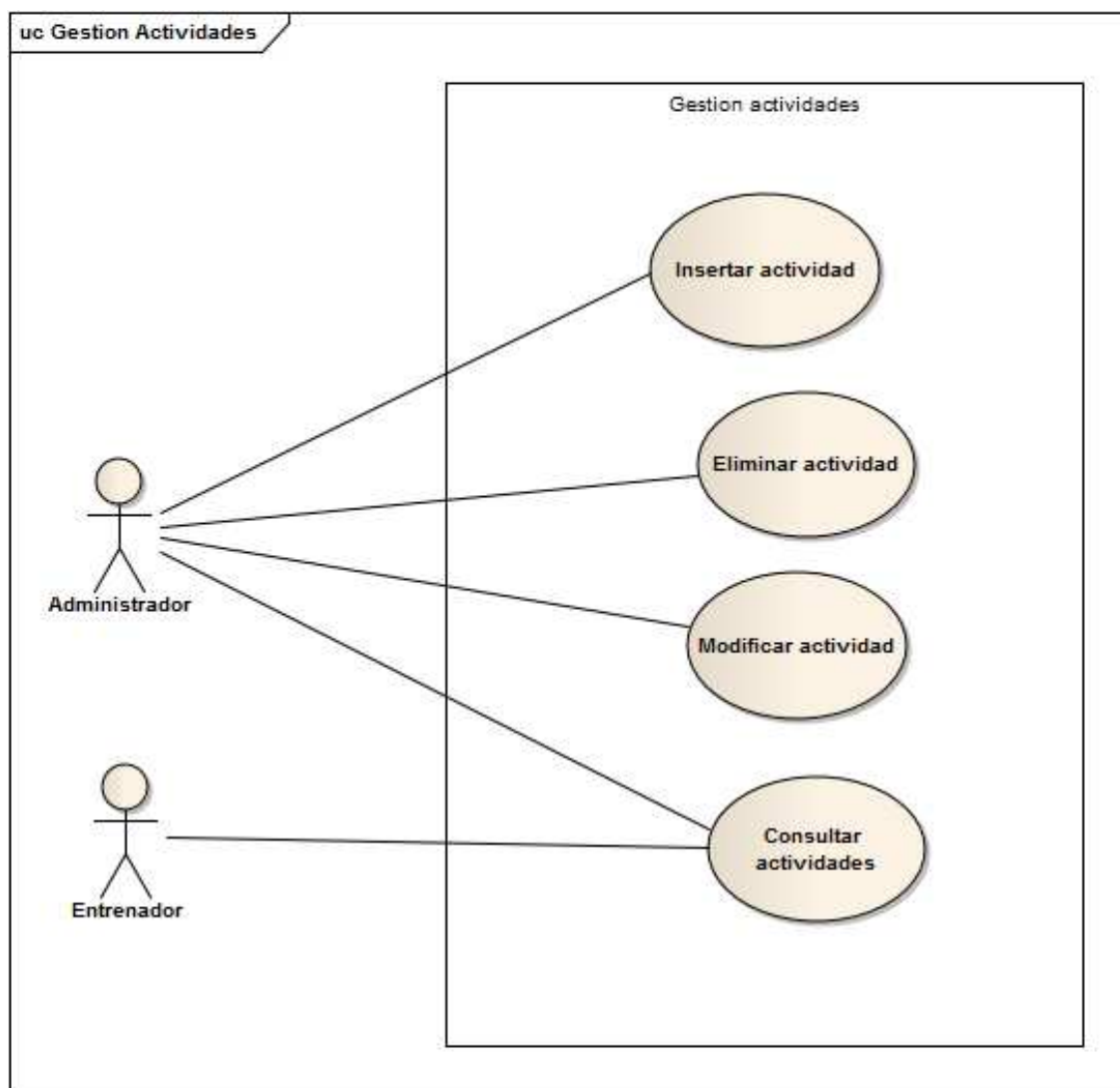


Diagrama de CU: Gestión de Grupos

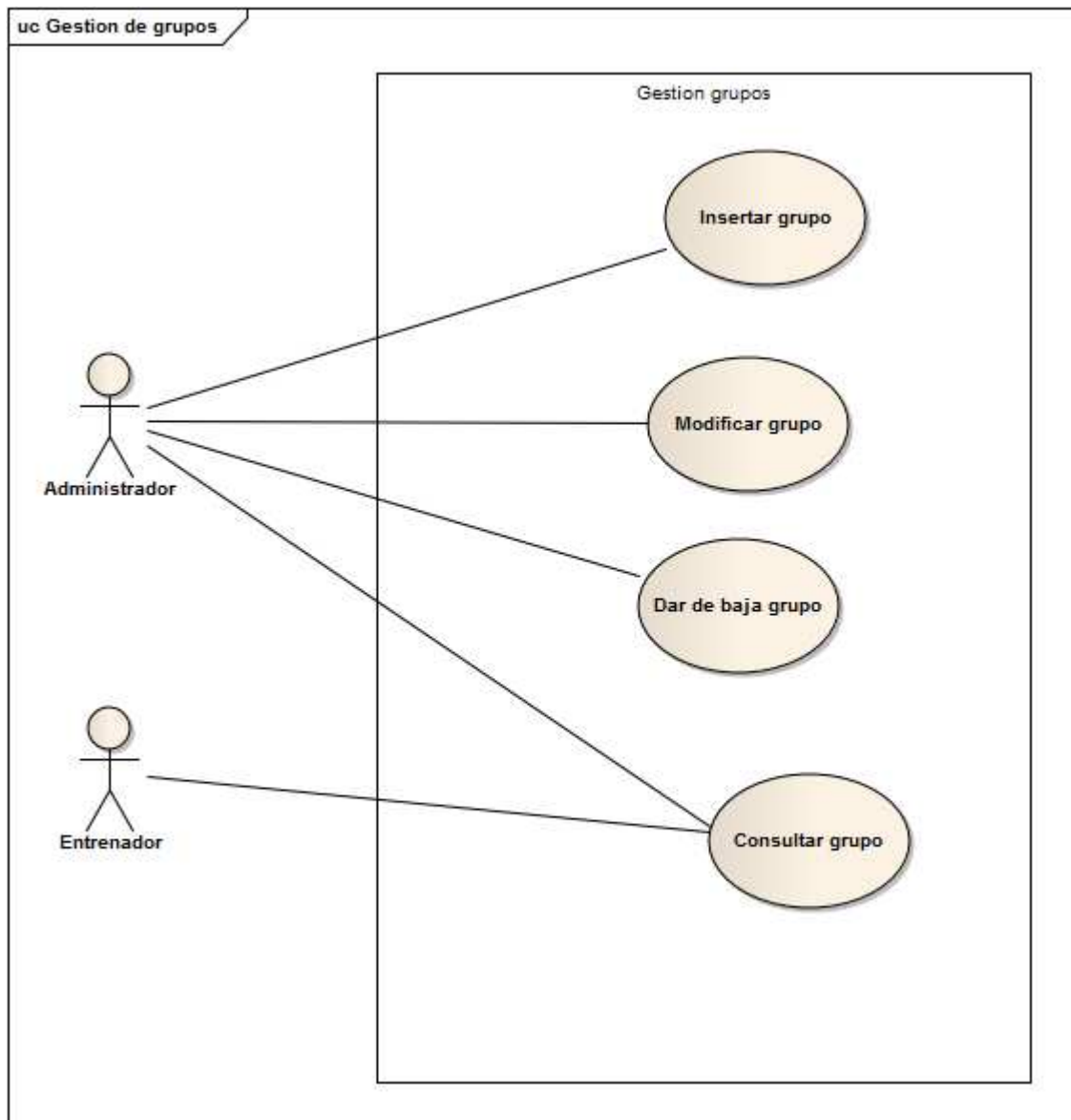


Diagrama de CU: Gestión de Temporadas

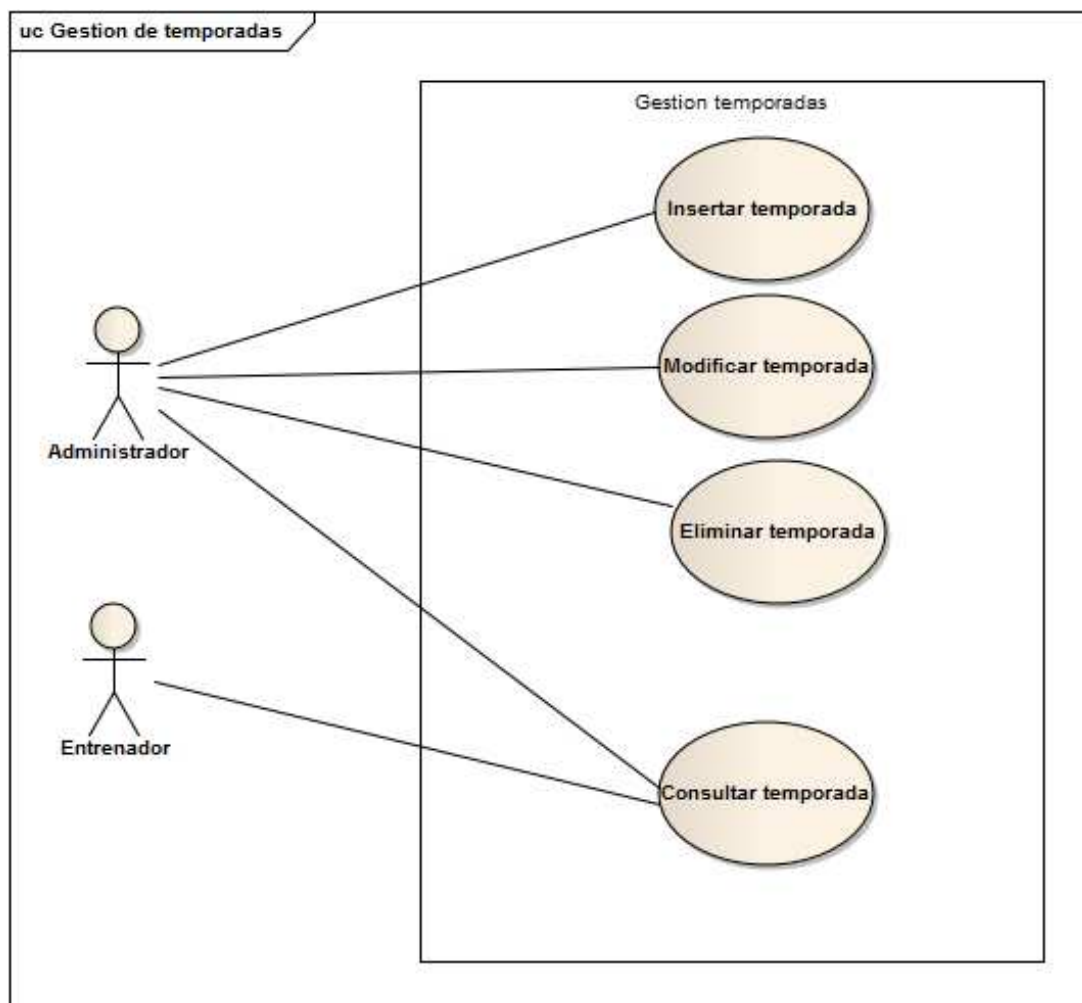


Diagrama de CU: Gestión de Equipos

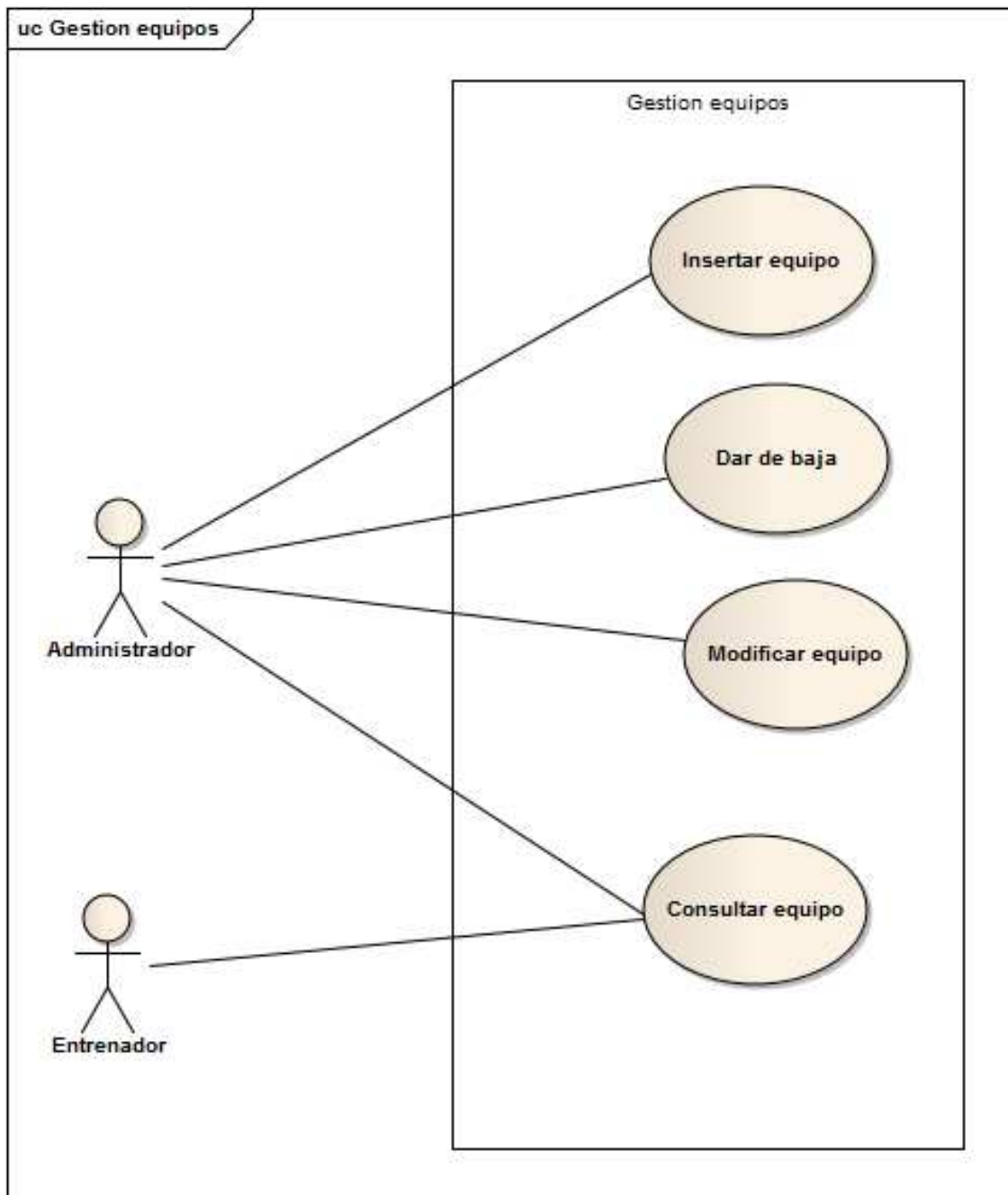


Diagrama de CU: Gestión de Instalaciones

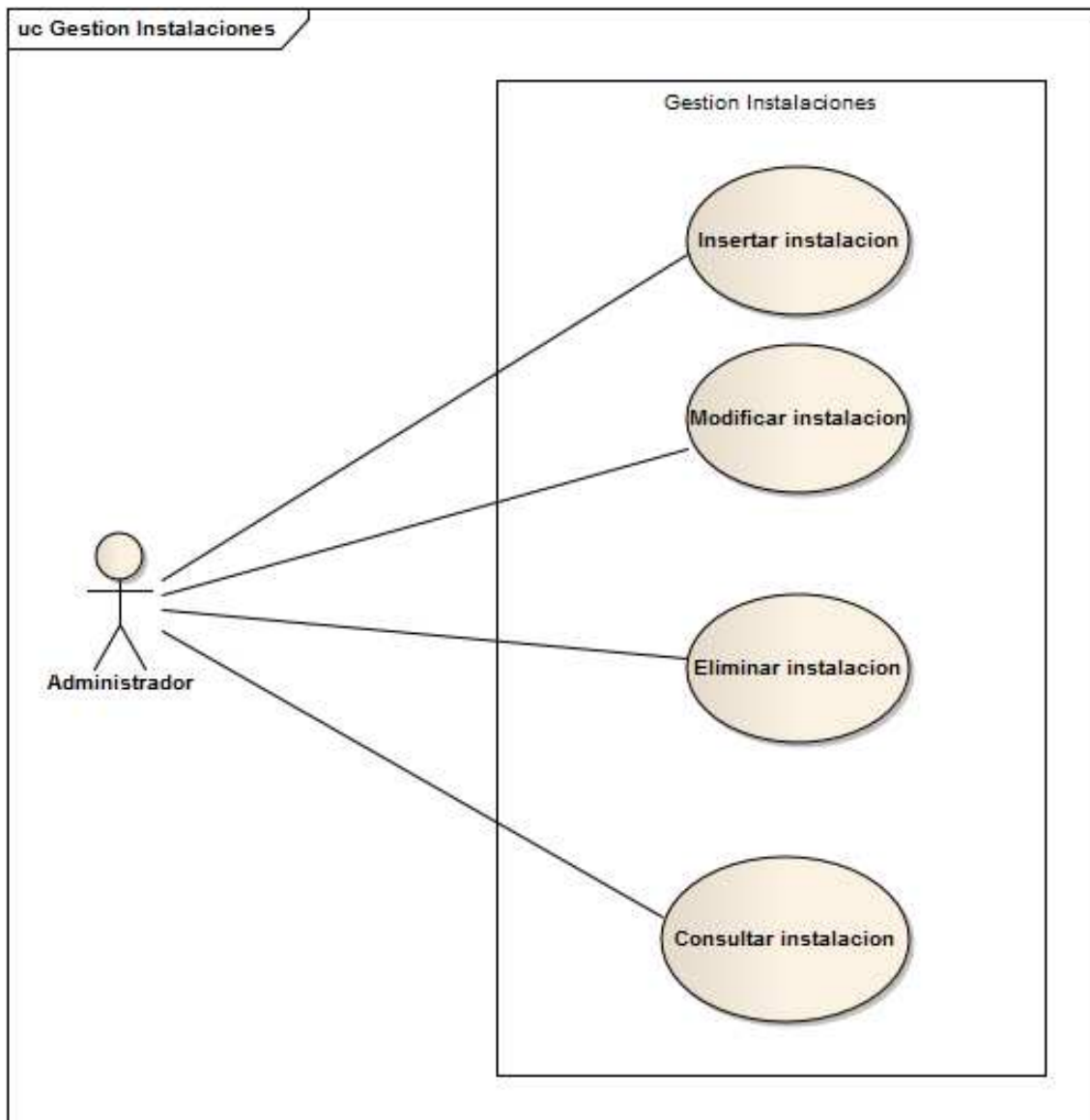


Diagrama de CU: Gestión de Pagos

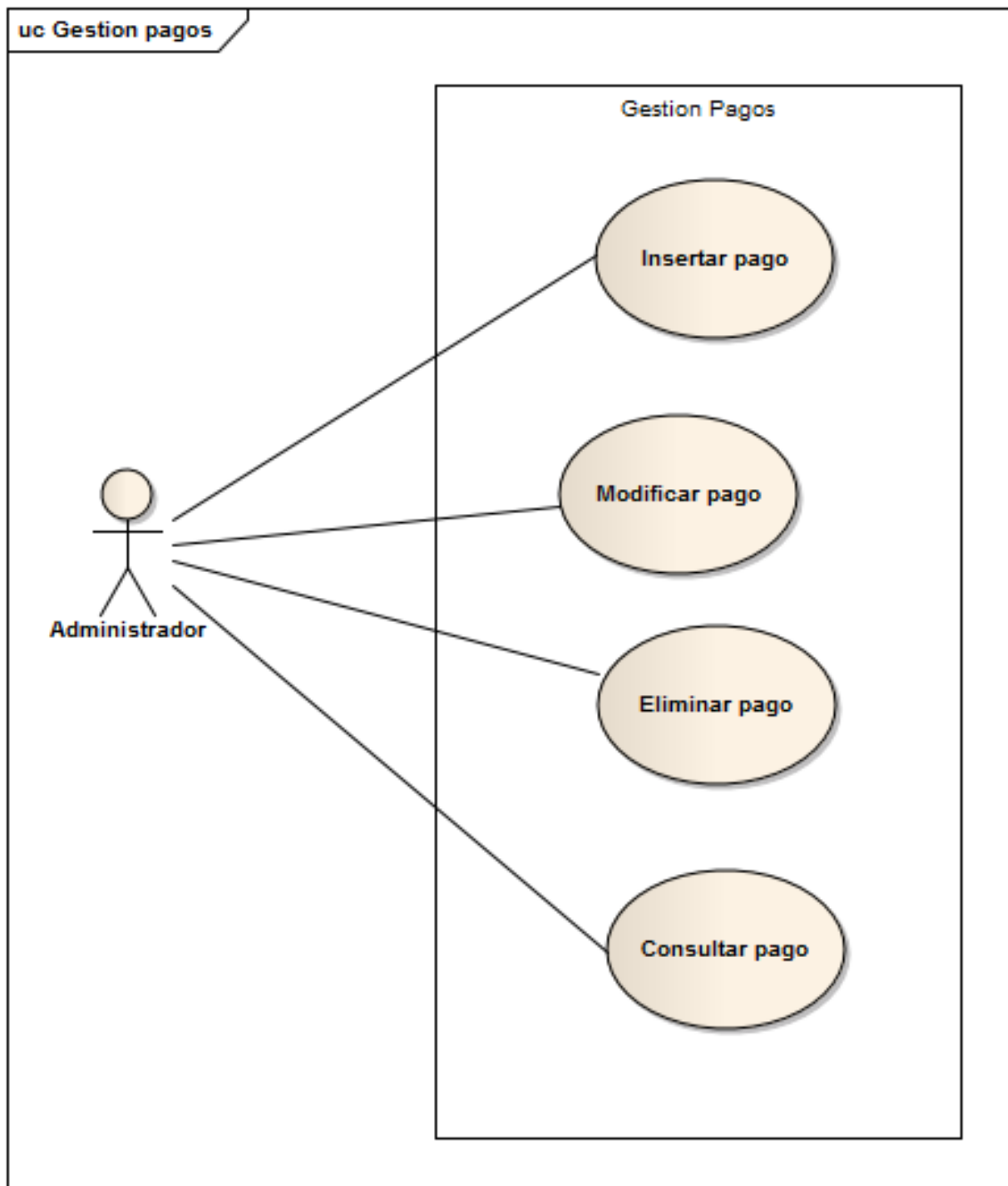
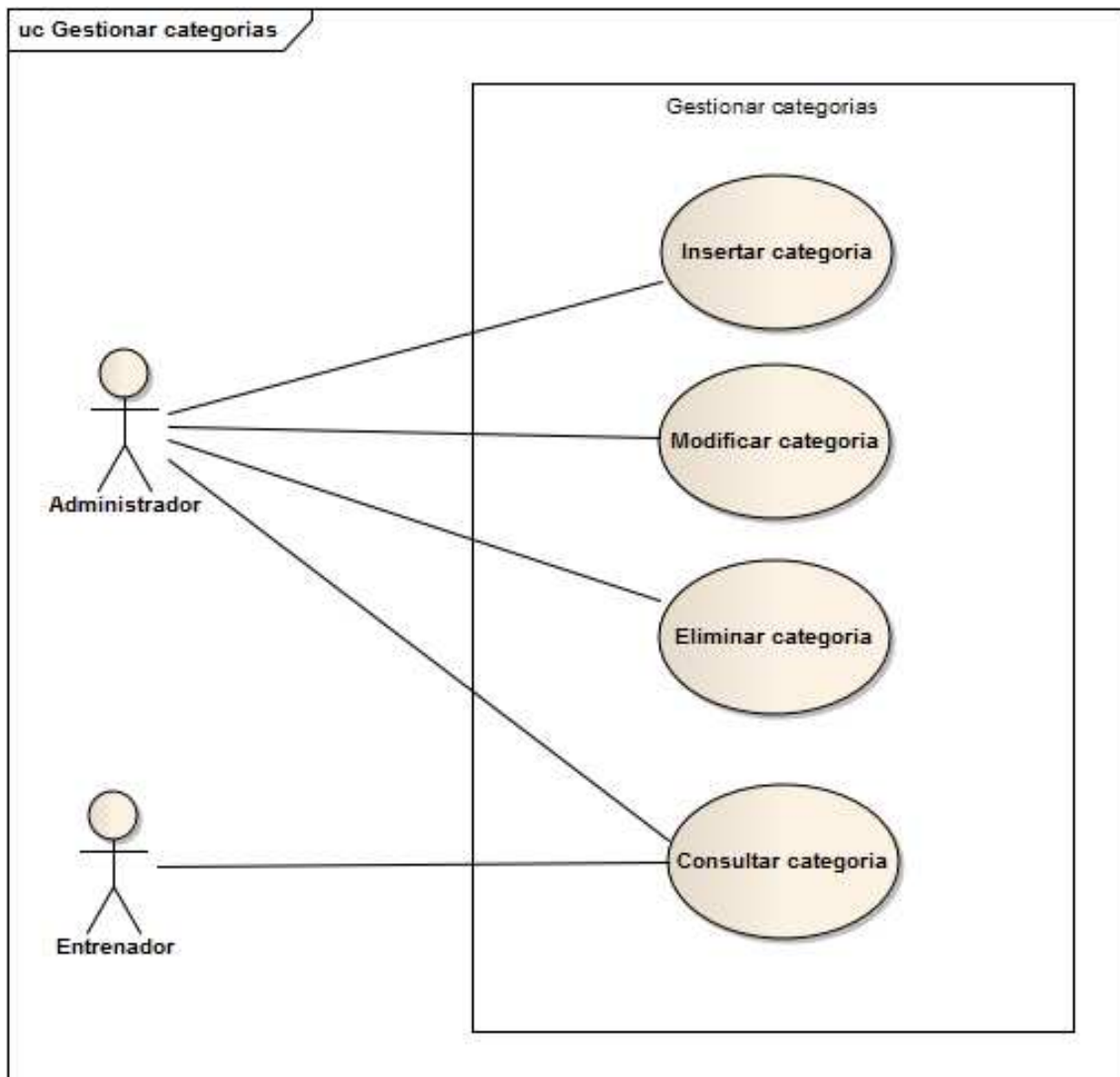


Diagrama de CU: Gestión de Categorías



Requisitos no funcionales

Estos requisitos son aquellos que describen cualidades o restricciones del sistema que no se relacionan de forma directa con el comportamiento funcional del mismo. A continuación se especifican los más importantes del sistema:

- No requiere un conocimiento específico para la utilización del software.
- La aplicación tendrá un manual de uso.
- La base de datos estará implementada en un lenguaje objeto relacional como mysql.
- La aplicación estará realizada en el lenguaje de programación Java.
- Se creara un script de tarea programada (cron) para la copia de seguridad de la base de datos debido a alguna inconsistencia del sistema.
- La interfaz debe reflejar claramente la distinción entre las distintas partes del sistema.
- La documentación del código fuente será llevada a cabo mediante la aplicación javadoc.
- El sistema se desplegara sobre una distribución Microsoft Windows.
- El administrador se encargara de tareas de mantenimiento.
- Cuando el sistema detecte una anomalía mostrara al usuario el mensaje de error pertinente y abortara la ejecución del proceso.
- El código fuente de la aplicación seguirá un estilo uniforme y normalizado para todos los módulos del mismo.
- El formato de las fechas será dd/mm/yy.
- El control de asistencia no es controlado, pero se puede incluir en la sección de observaciones de los alumnos.

Operaciones del sistema – Diagramas de secuencia de sistema

Se van a describir diagramas de secuencia de las operaciones del sistema.

Gestión de Actividades

Diagrama de secuencia:Insertar Actividad

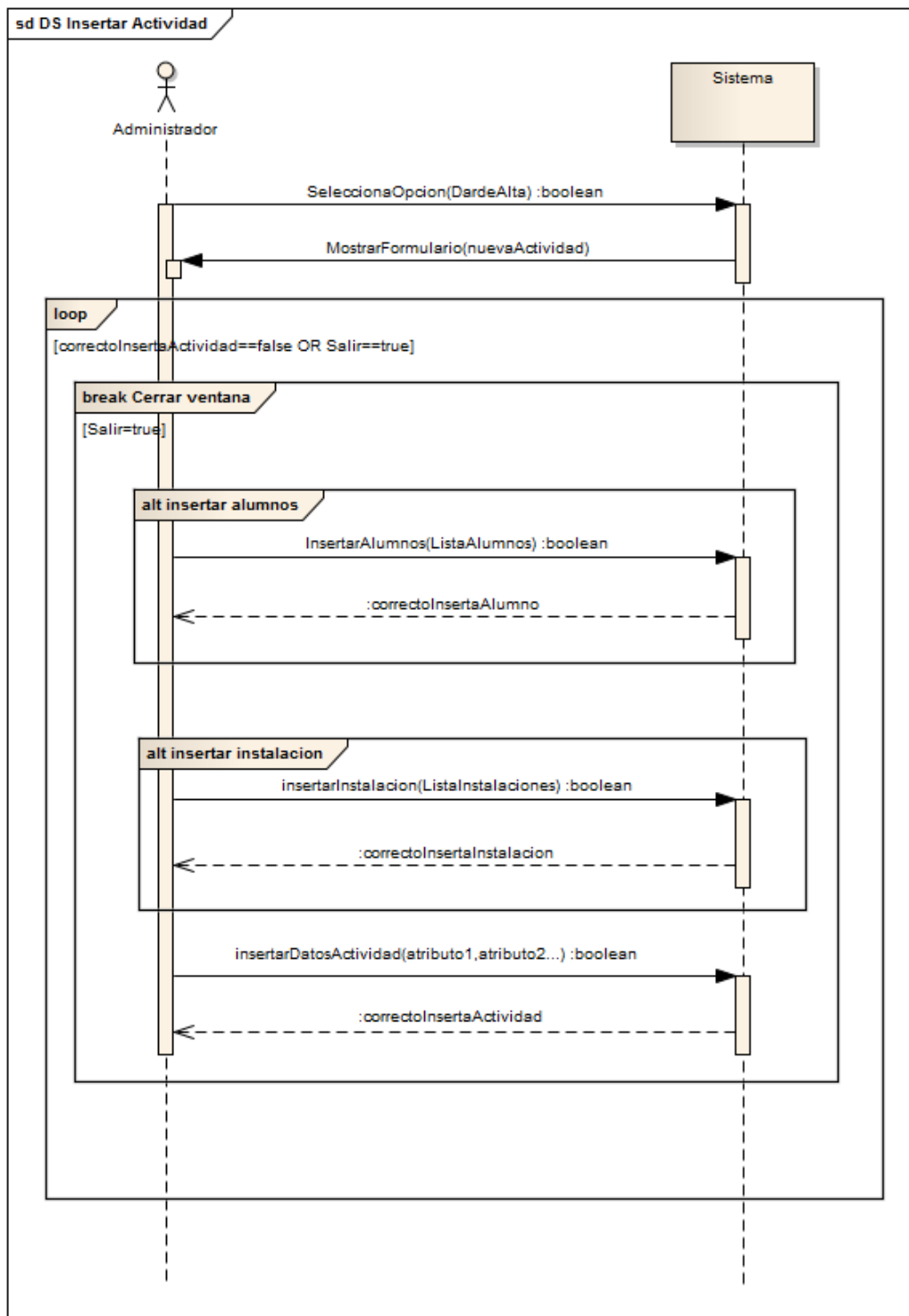


Diagrama de secuencia: Consultar Actividad

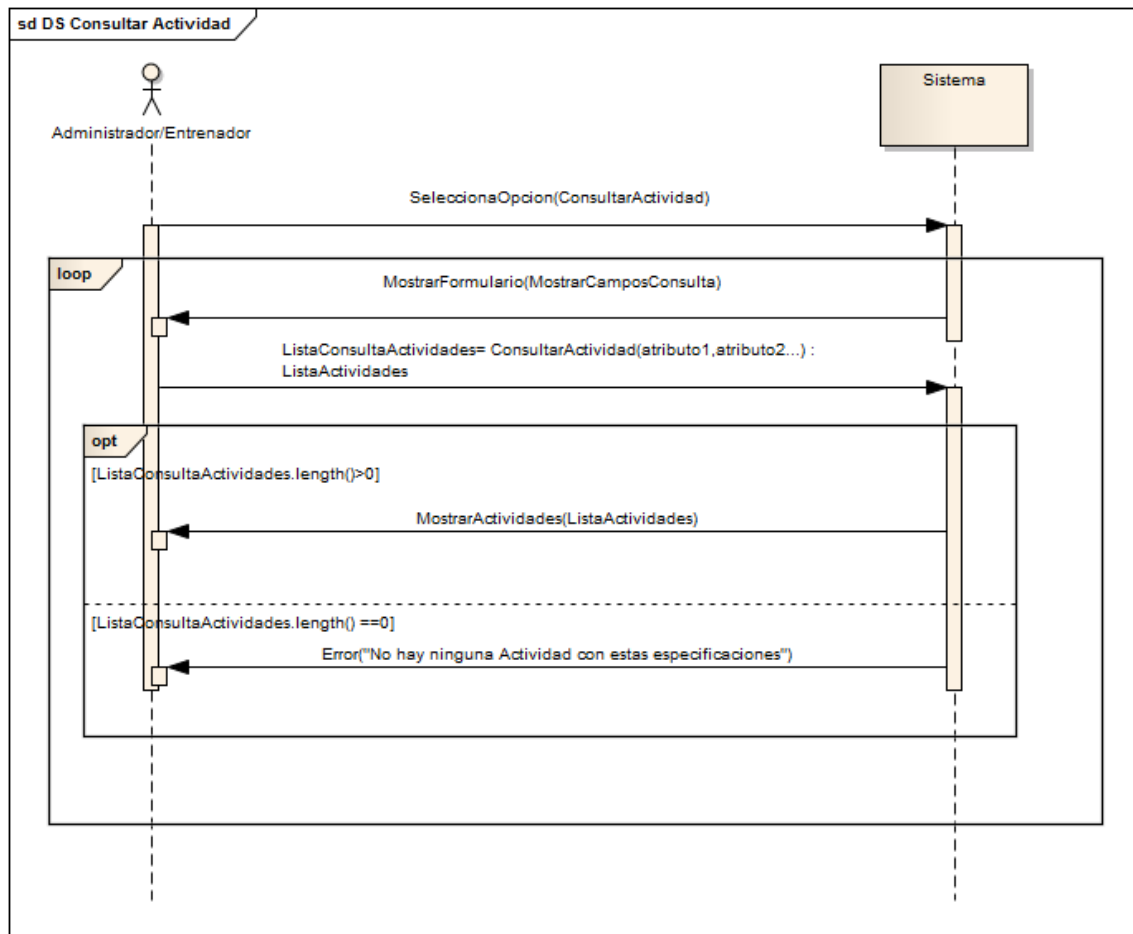


Diagrama de secuencia: Eliminar Actividad

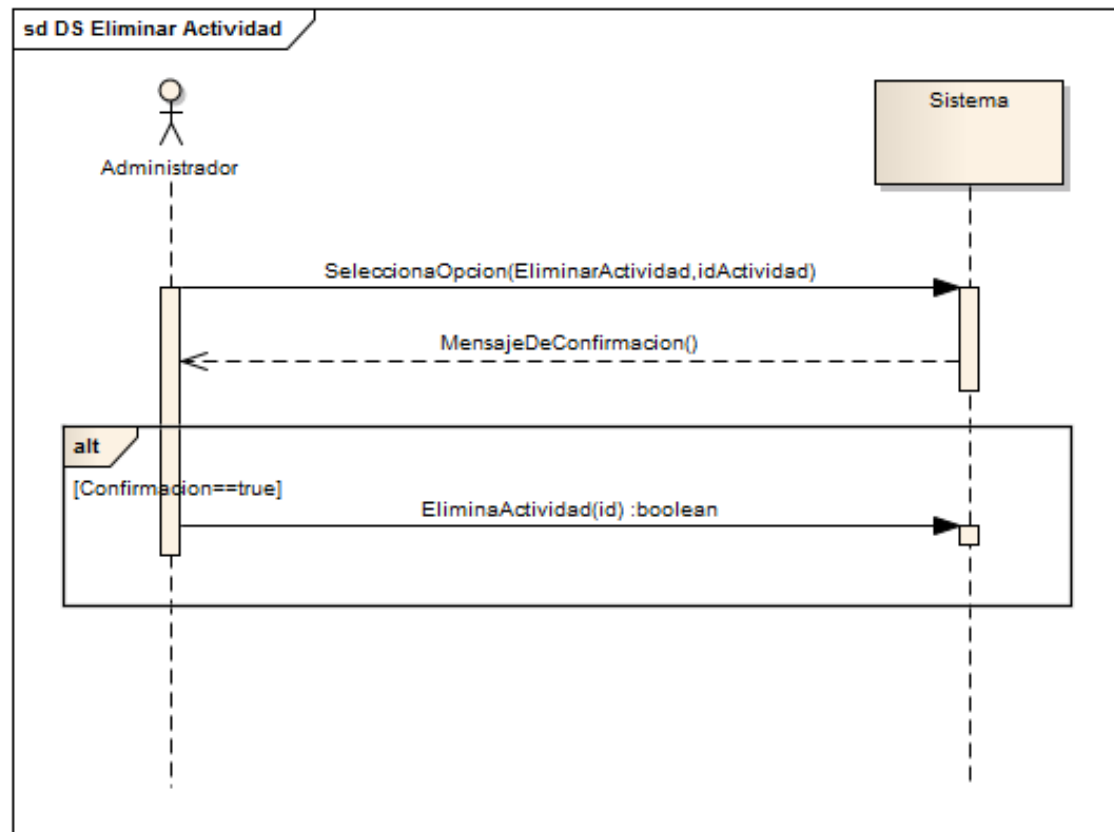
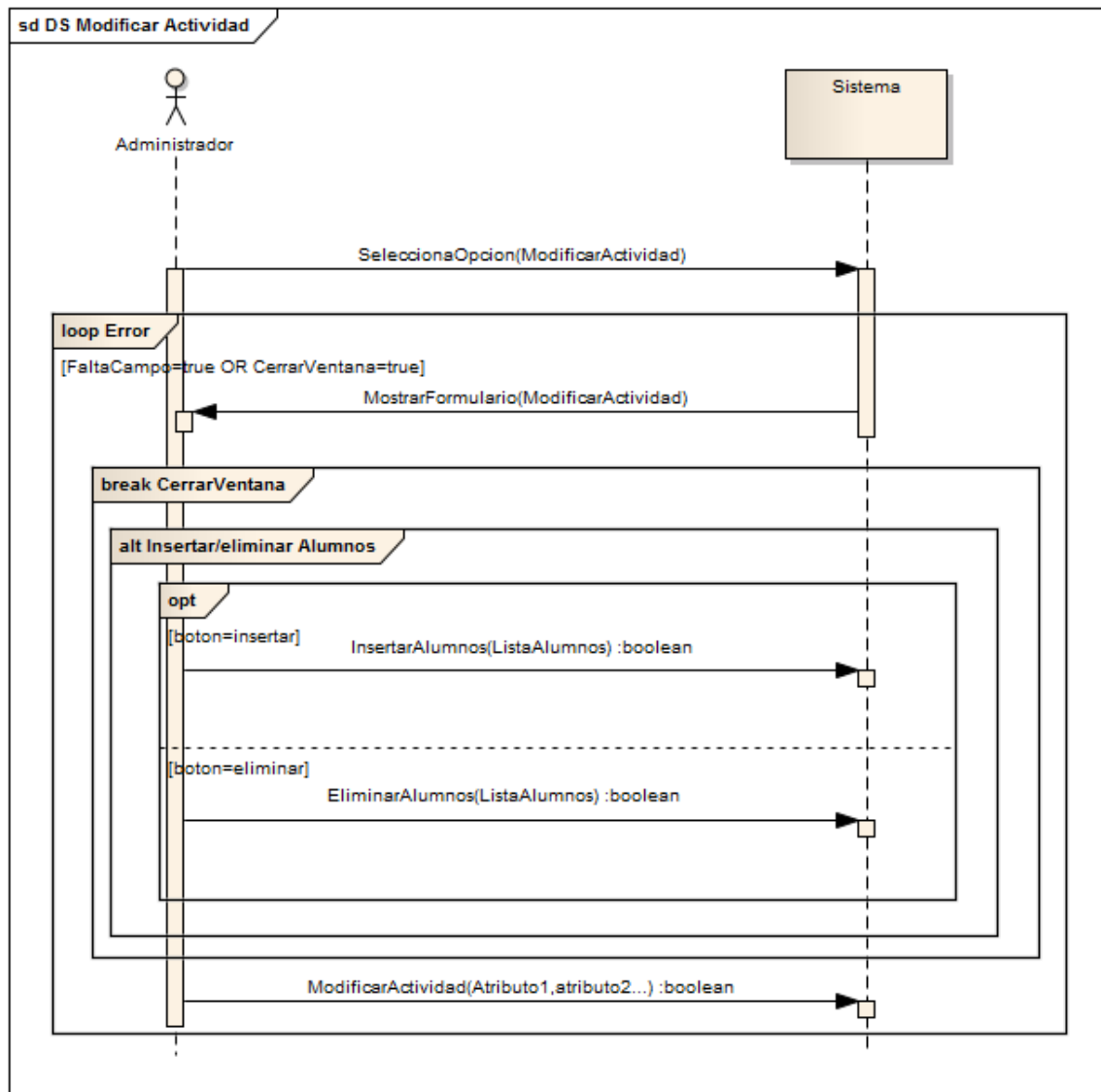


Diagrama de secuencia: ModificarActividad



Gestión de Categorías

Diagrama de secuencia: Insertar Categoría

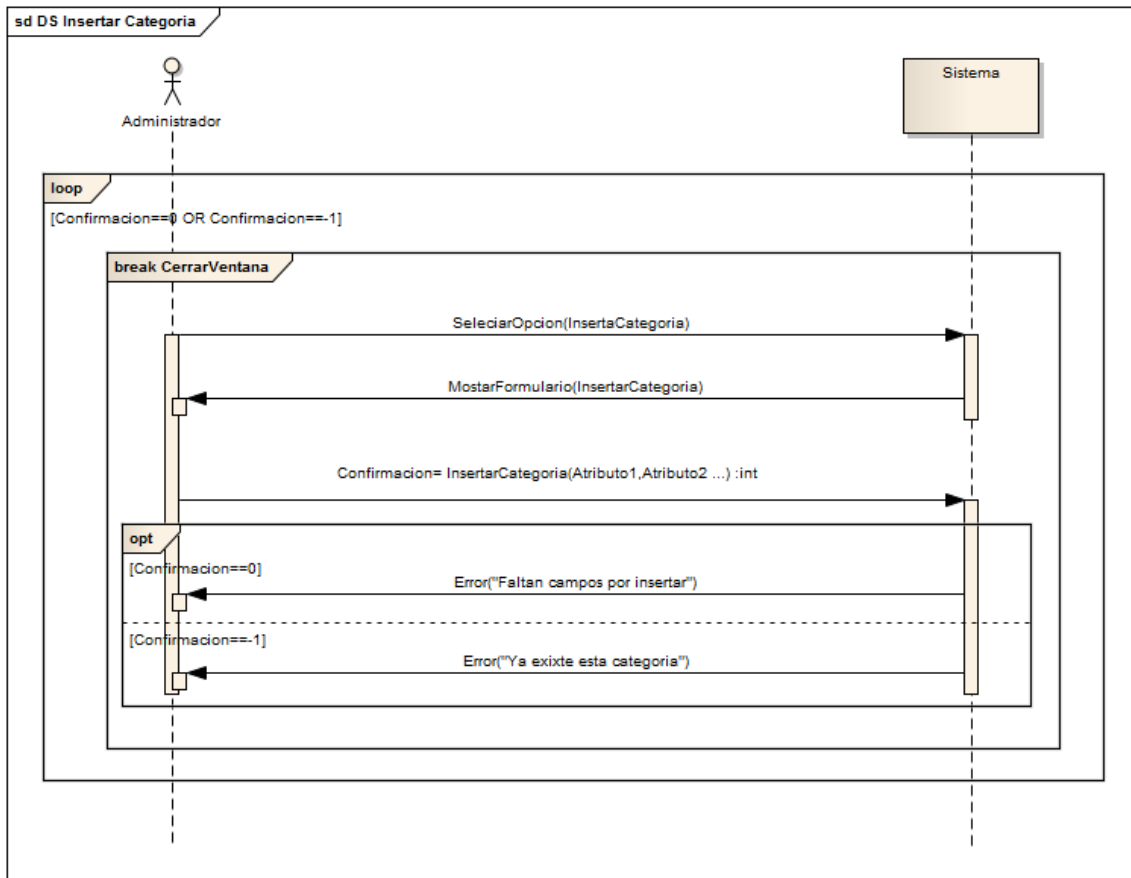


Diagrama de secuencia: Consultar Categoría

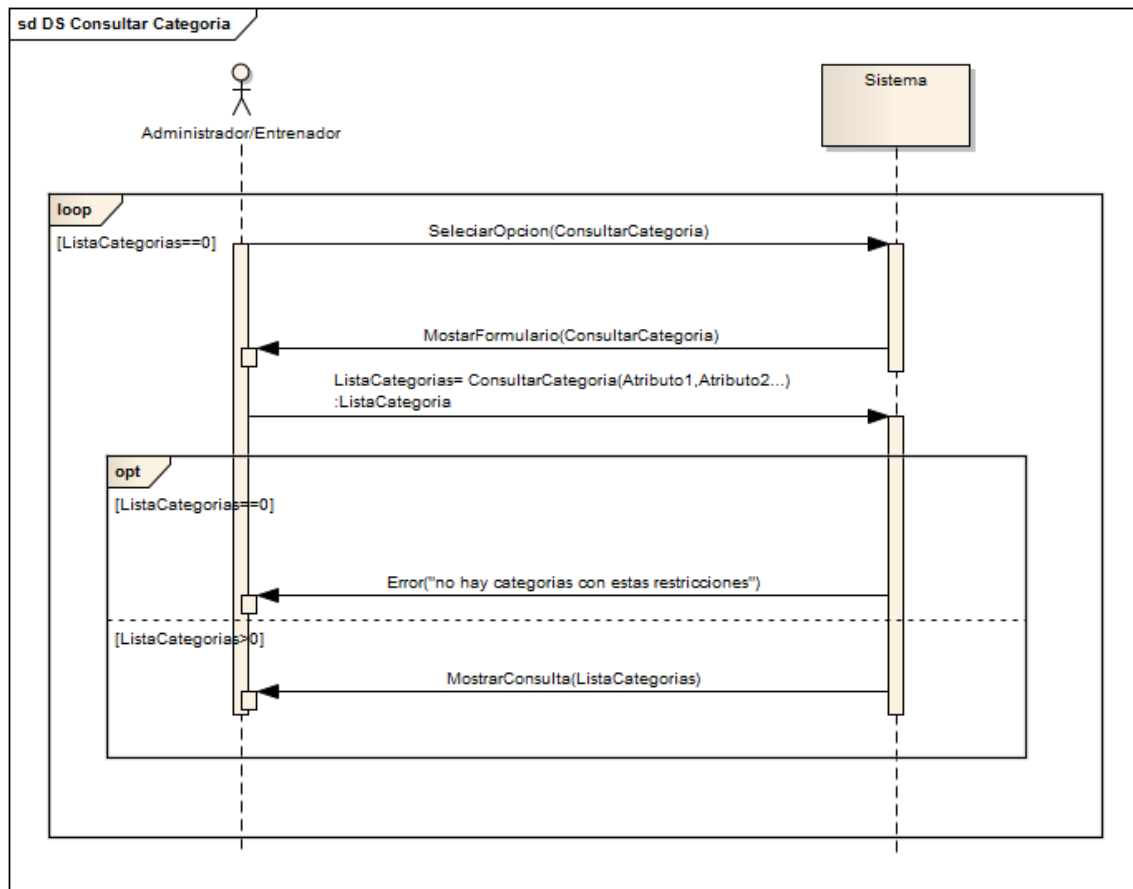


Diagrama de secuencia: Eliminar Categoría

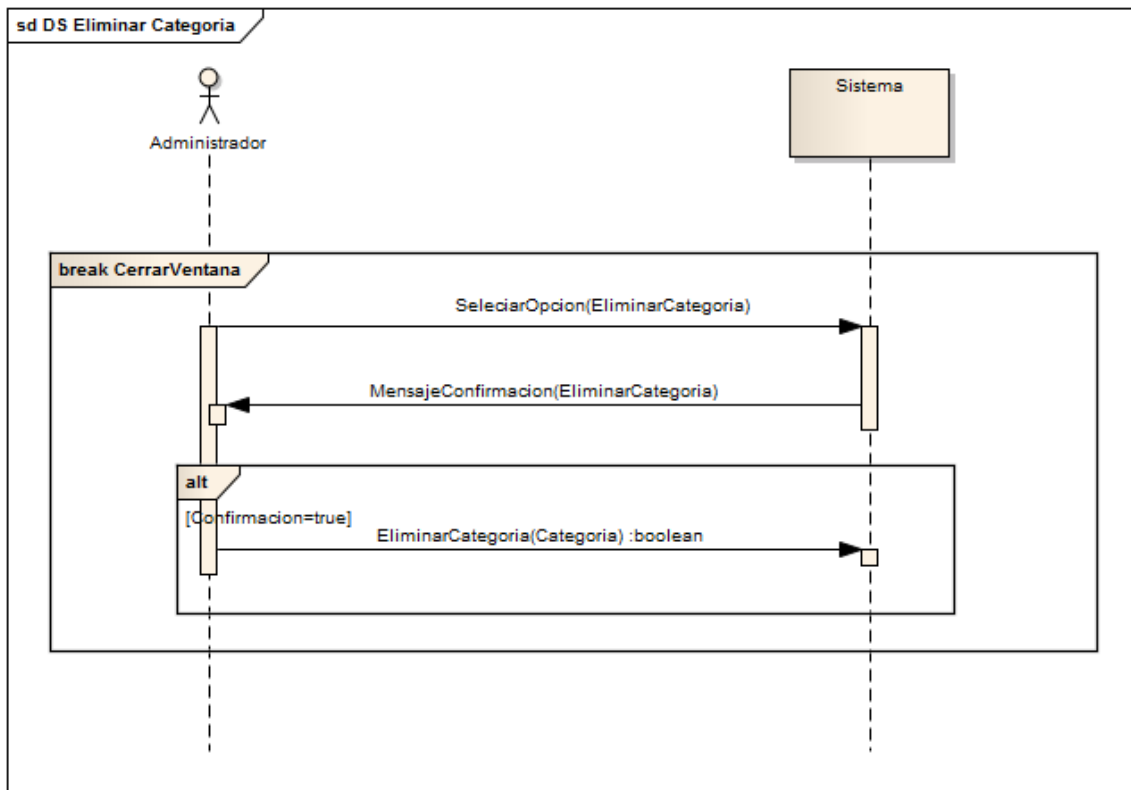
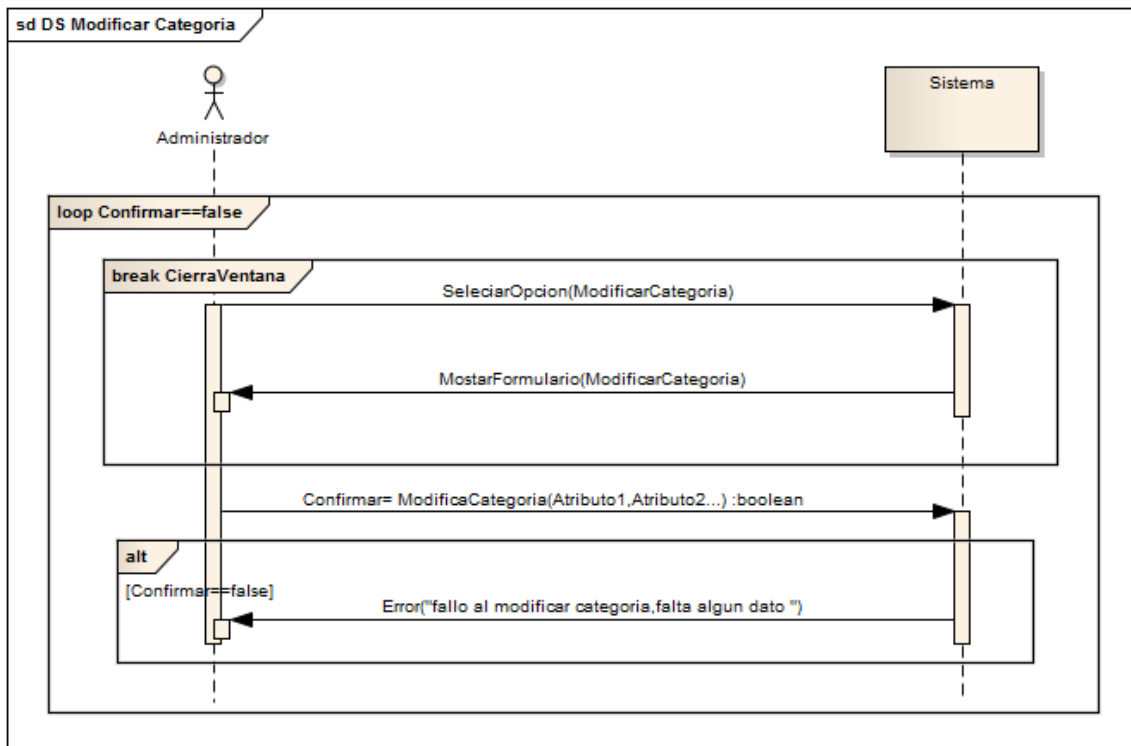


Diagrama de secuencia: Modificar Categoría



Gestión de Equipos

Diagrama de secuencia: Insertar Equipo

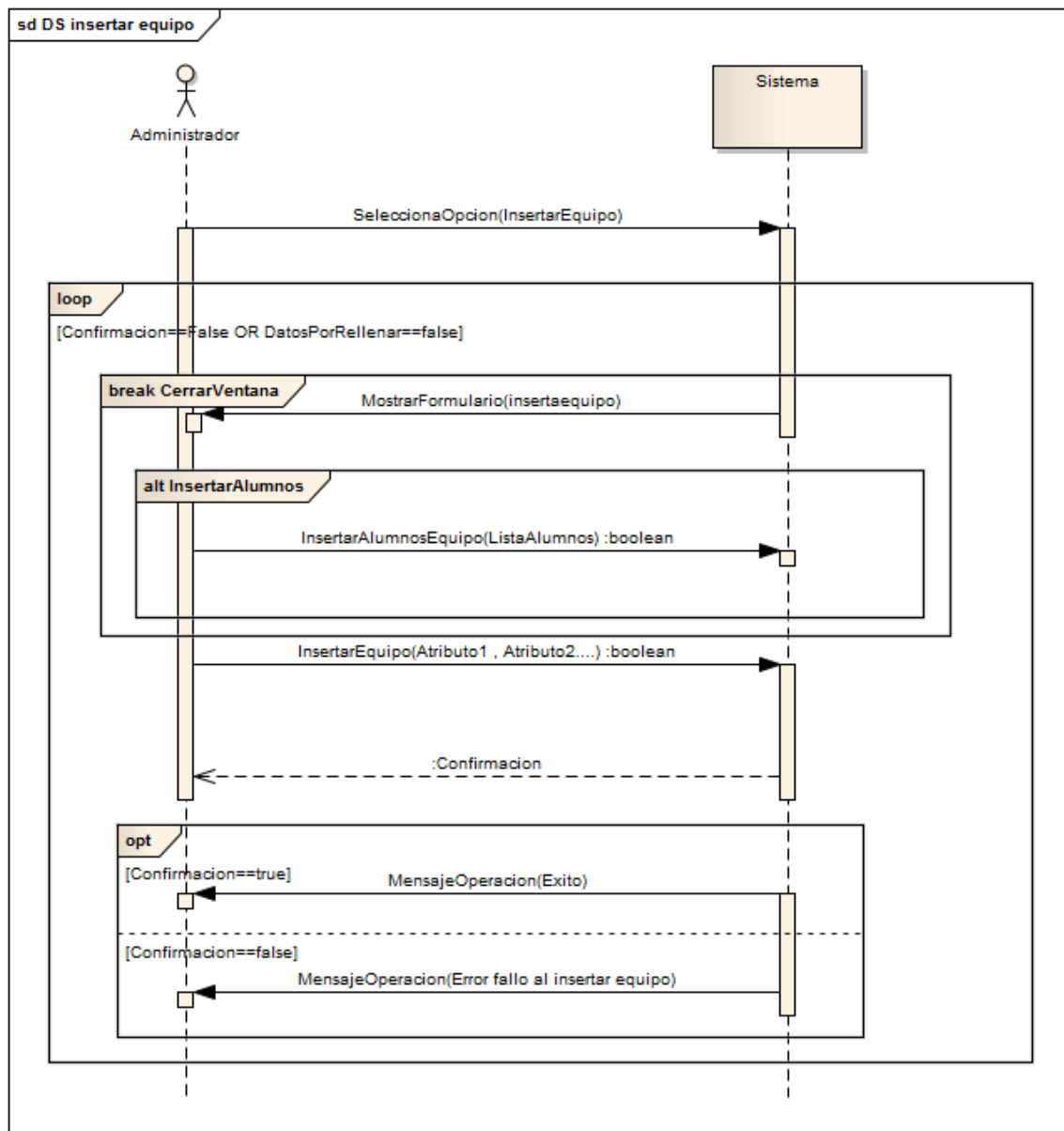


Diagrama de secuencia: Consultar Equipo

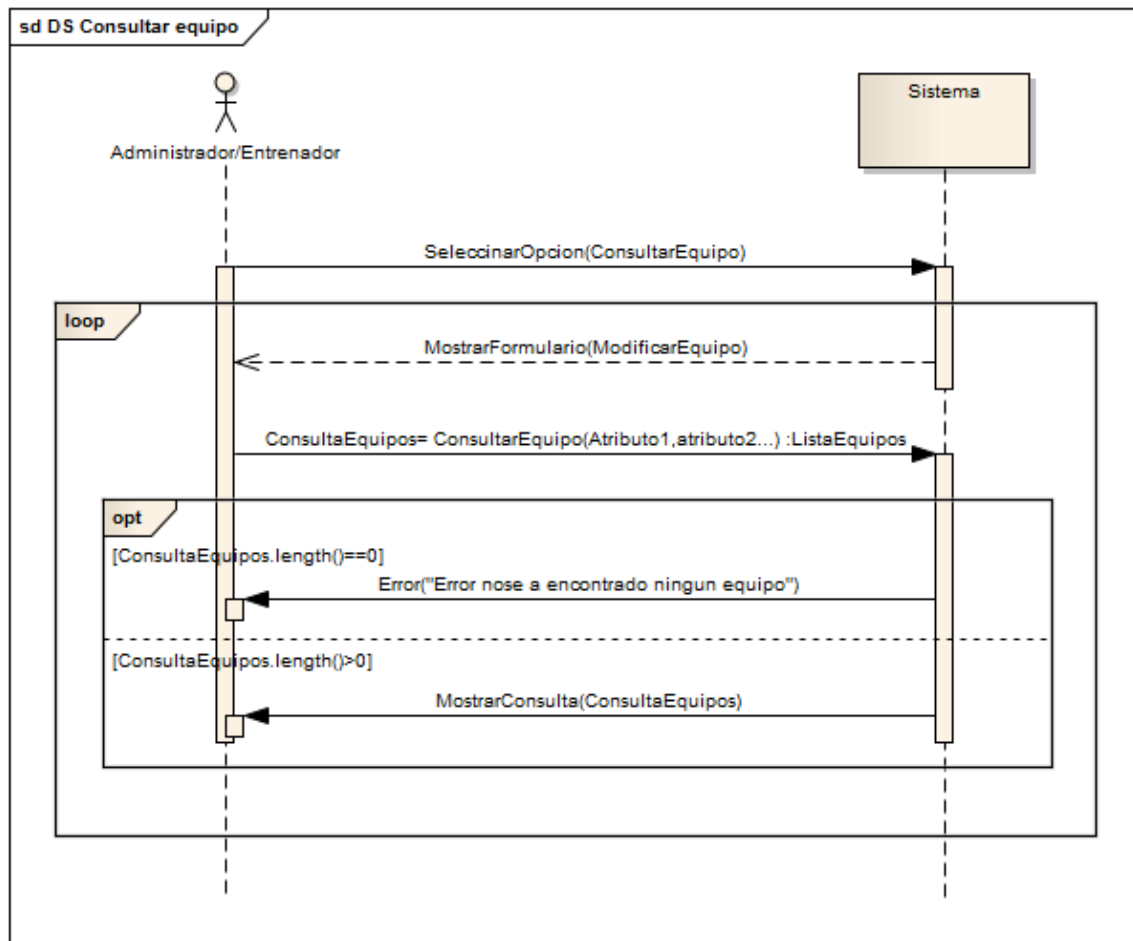


Diagrama de secuencia: Eliminar Equipo

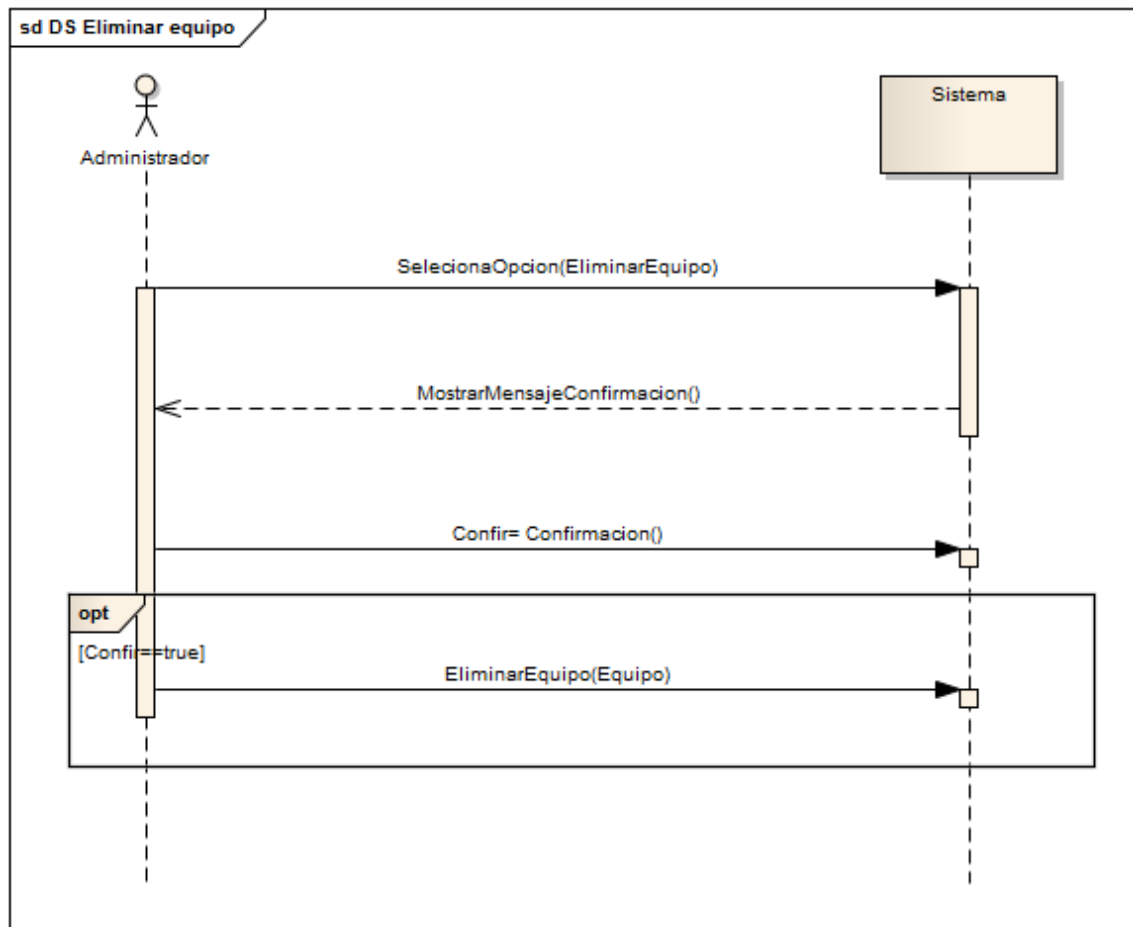
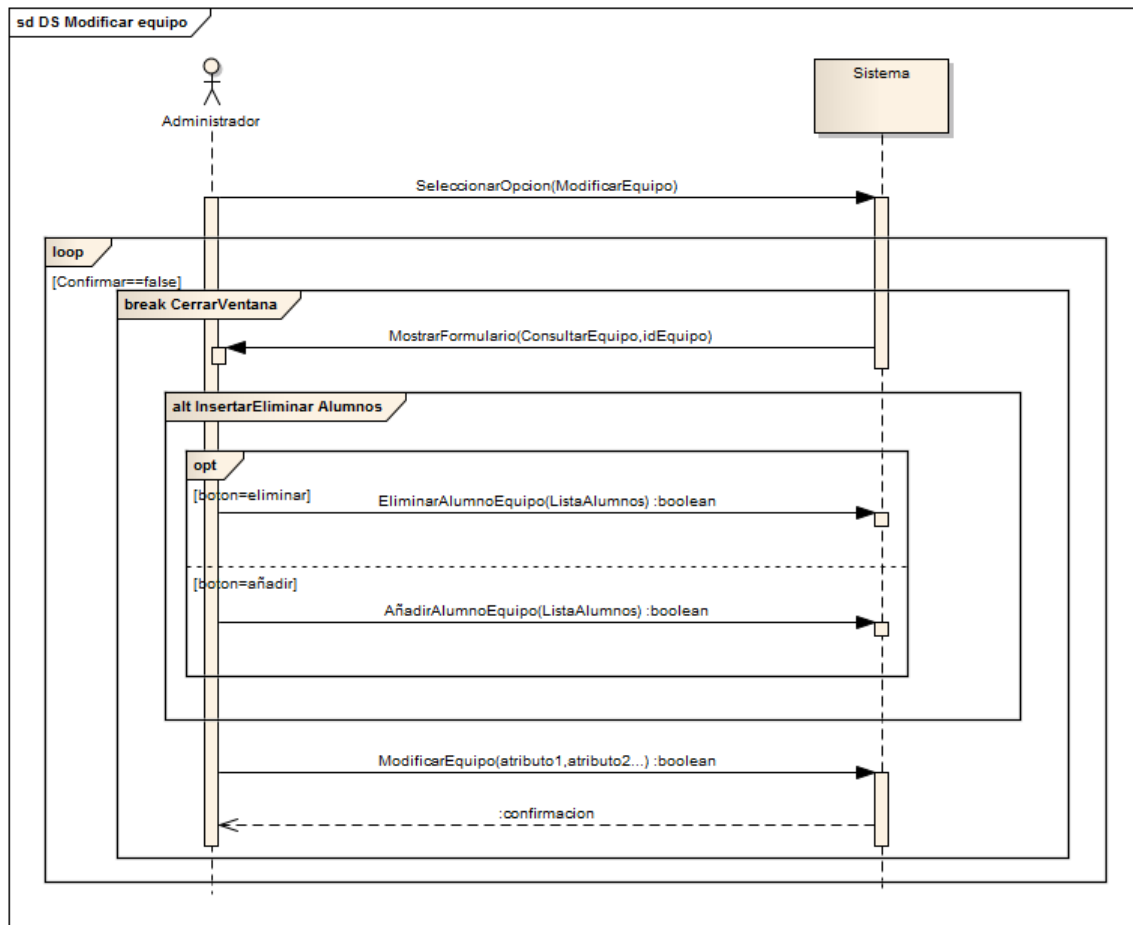


Diagrama de secuencia: Modificar Equipo



Gestión de Grupos

Diagrama de secuencia: Insertar Grupo

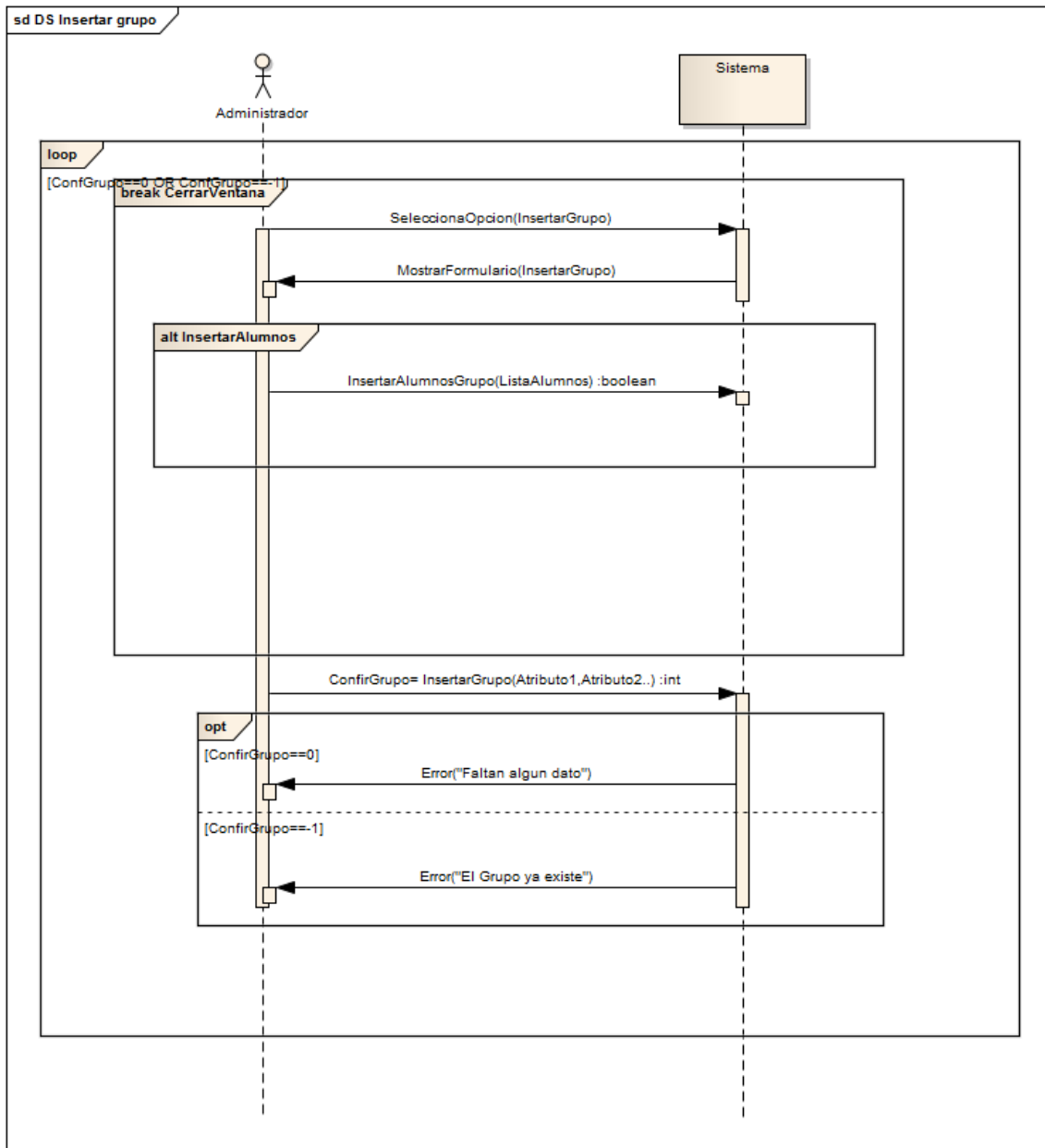


Diagrama de secuencia: Consultar Grupo

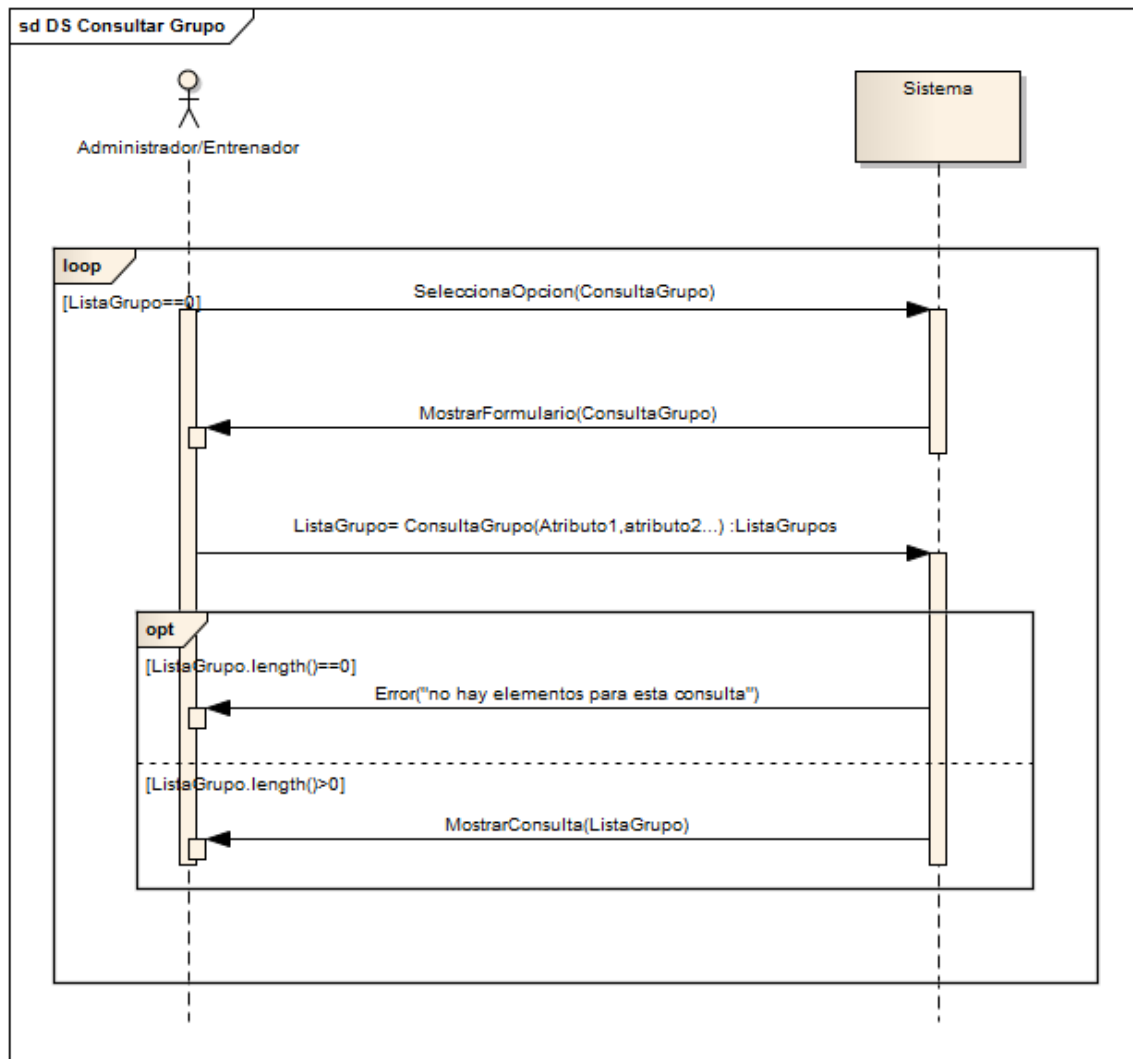


Diagrama de secuencia: Eliminar Grupo

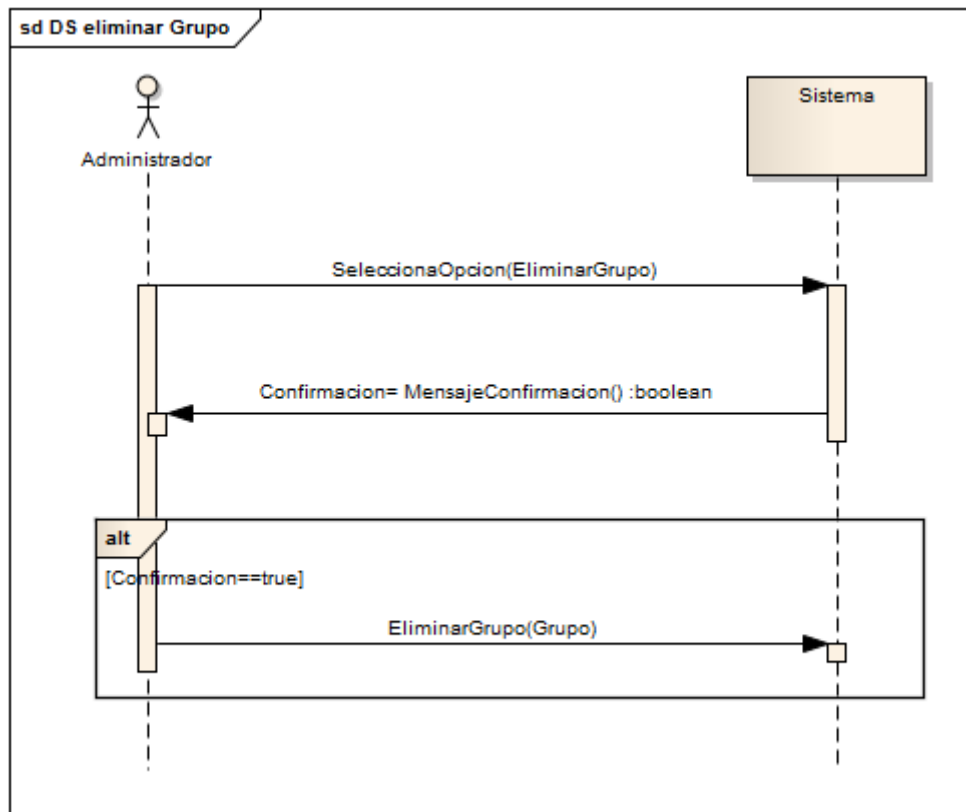
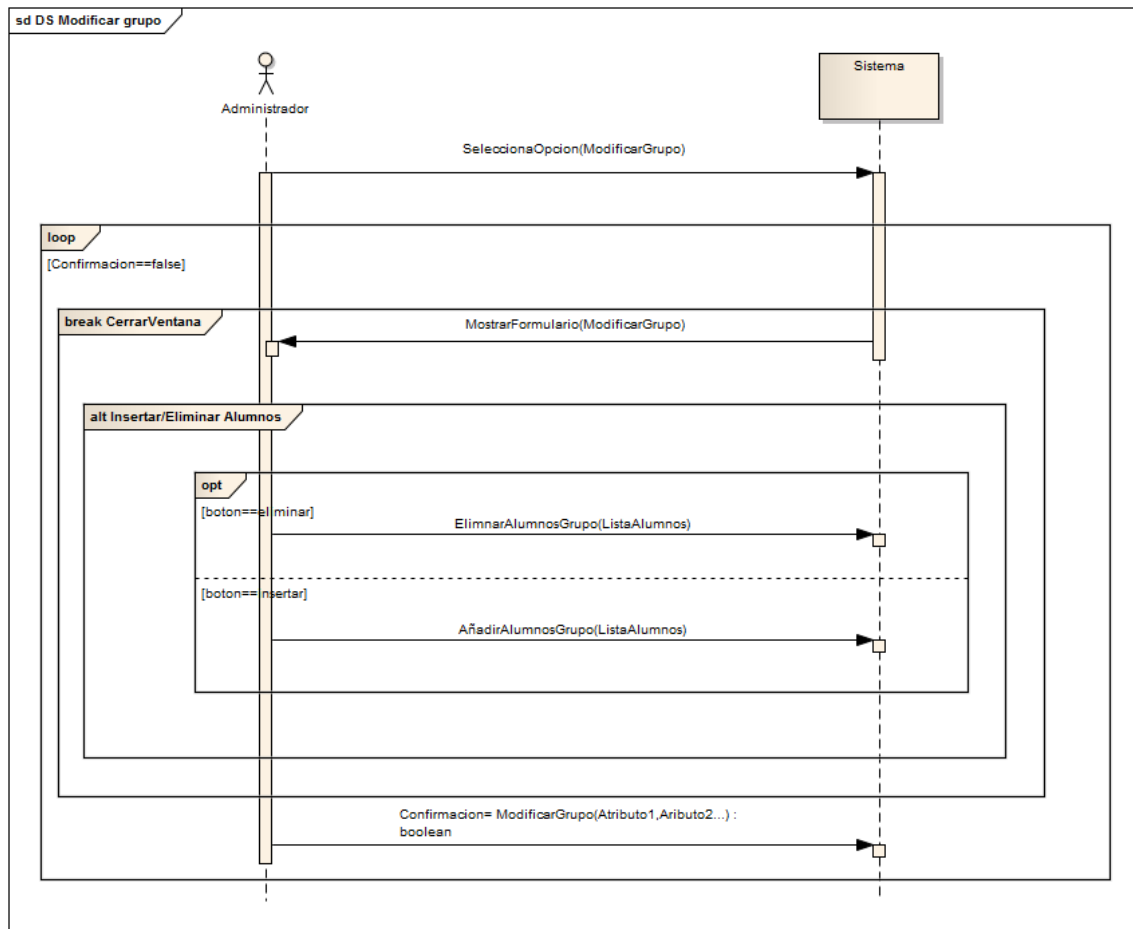


Diagrama de secuencia: Modificar Grupo



Gestión de Instalaciones

Diagrama de secuencia: Insertar Instalaciones

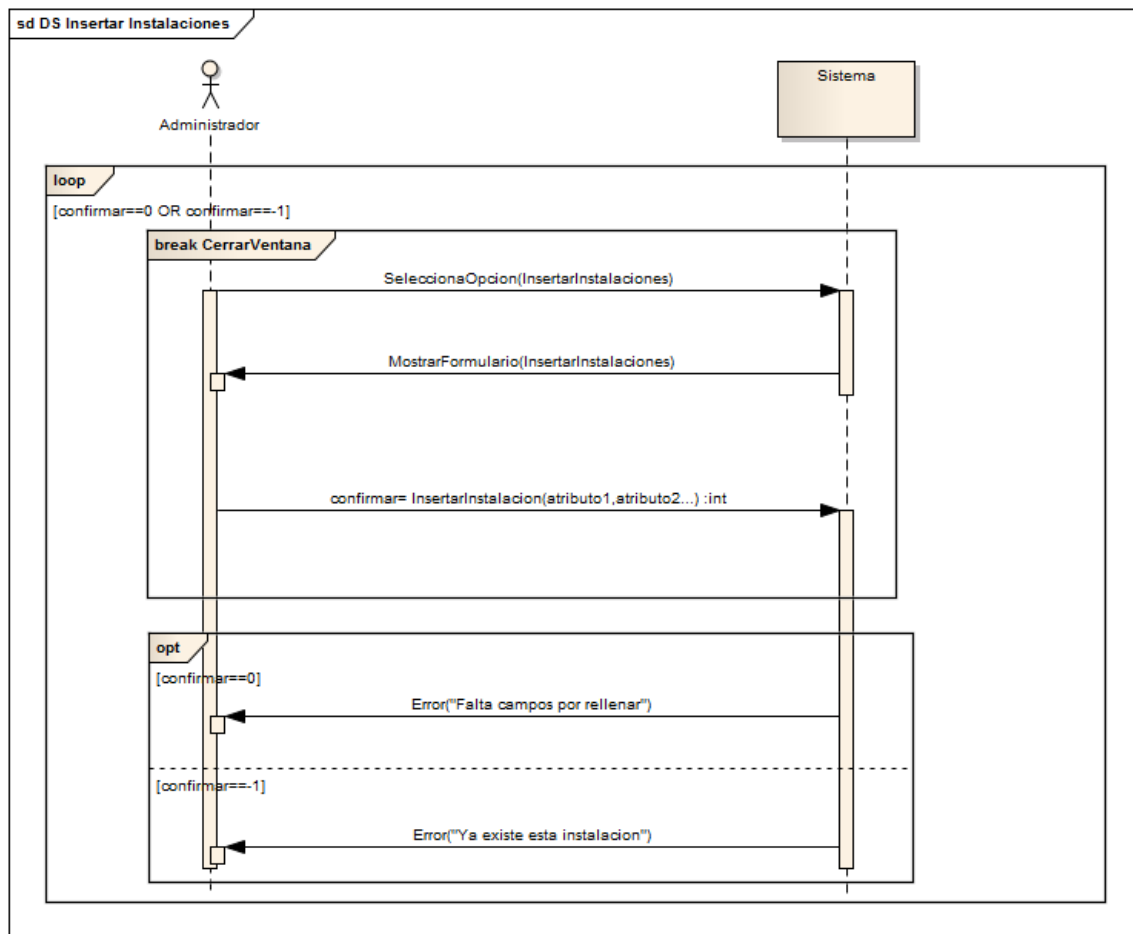


Diagrama de secuencia: Consultar Instalaciones

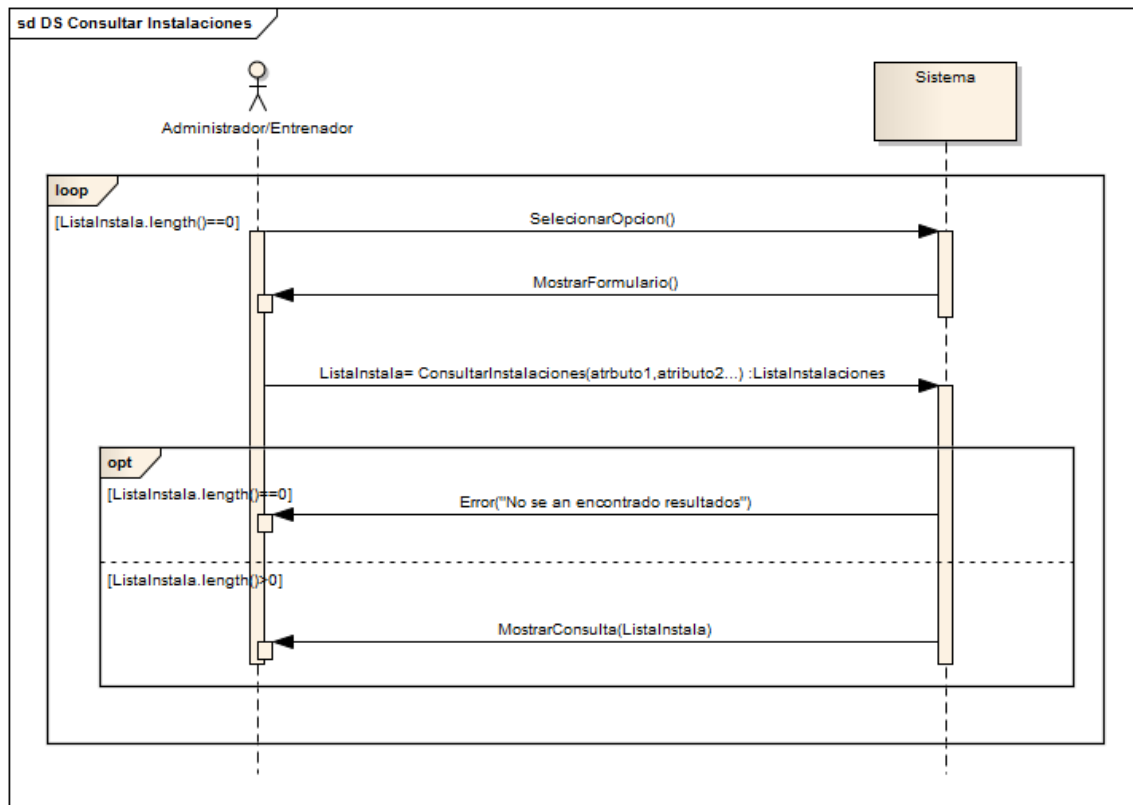


Diagrama de secuencia: *Eliminar Instalaciones*

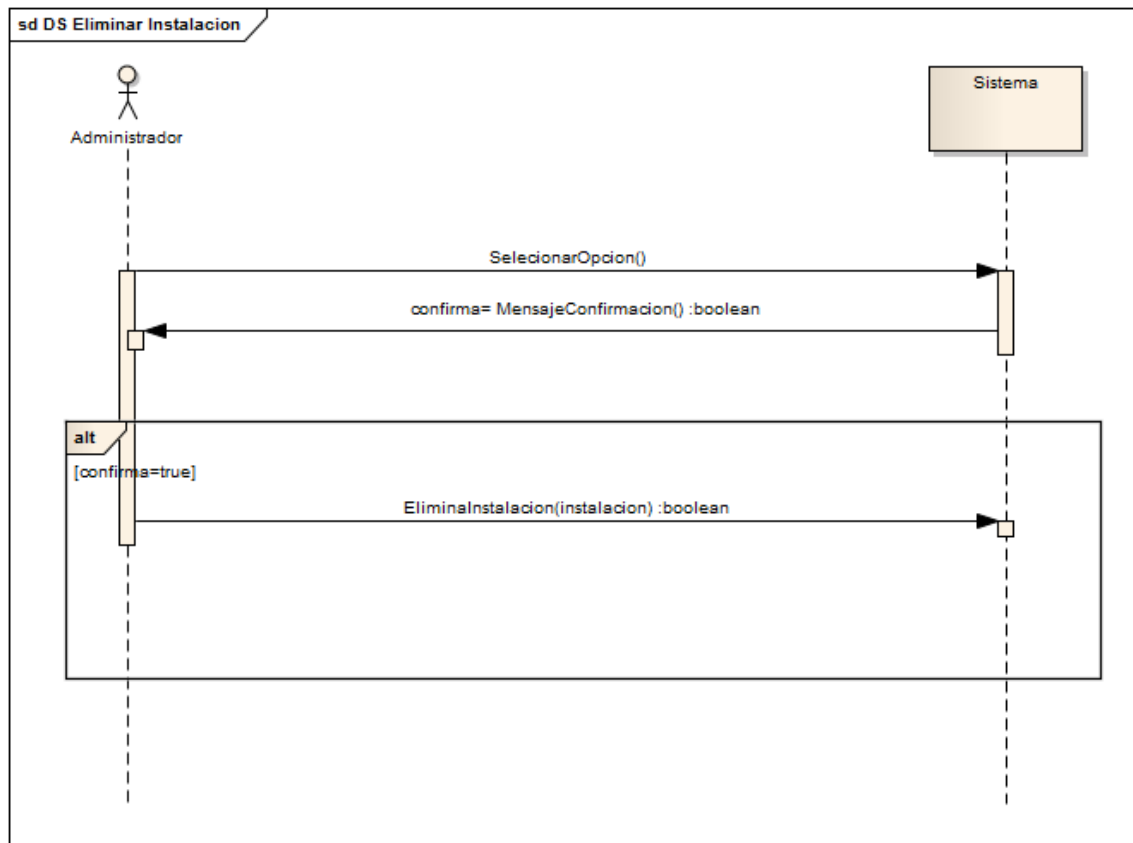
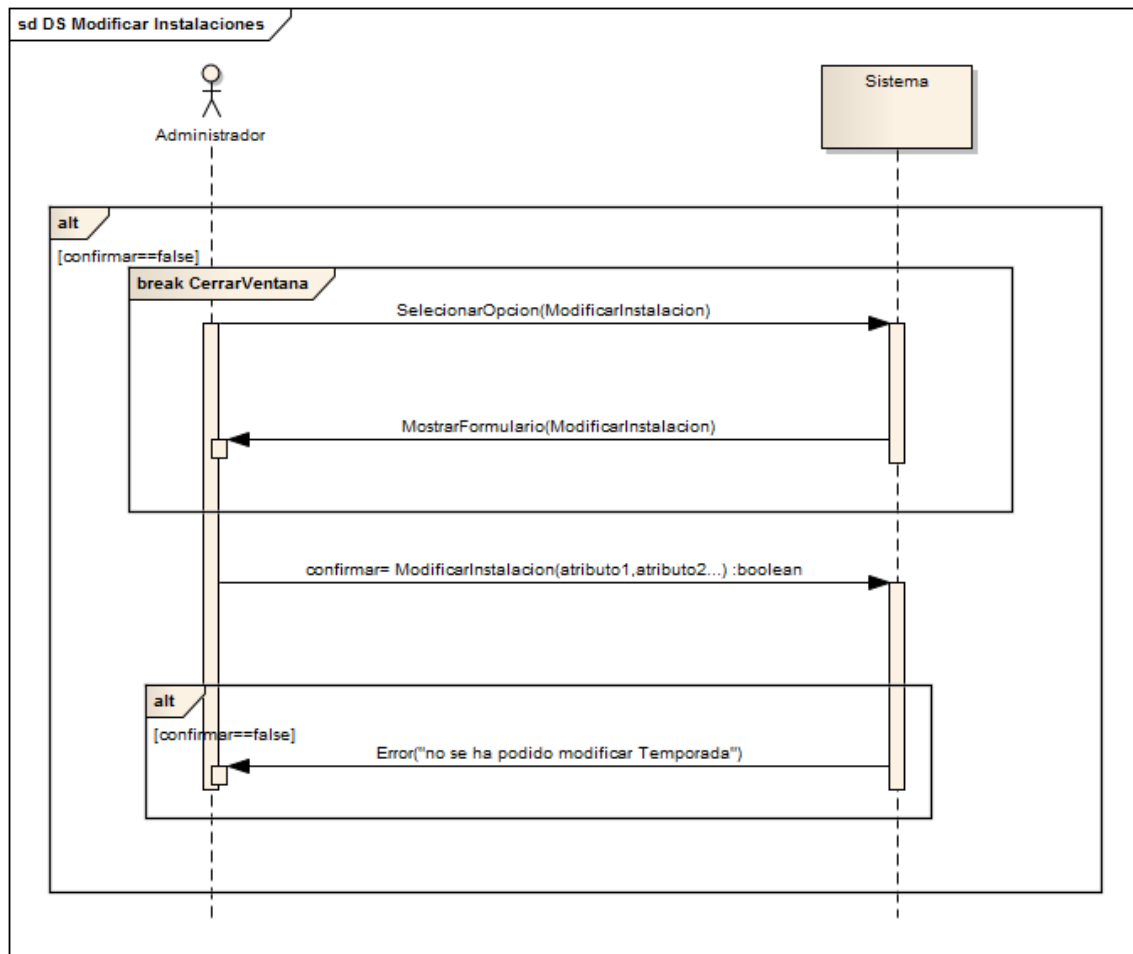


Diagrama de secuencia: Modificar Instalaciones



Gestión de Pagos

Diagrama de secuencia: Insertar Pago

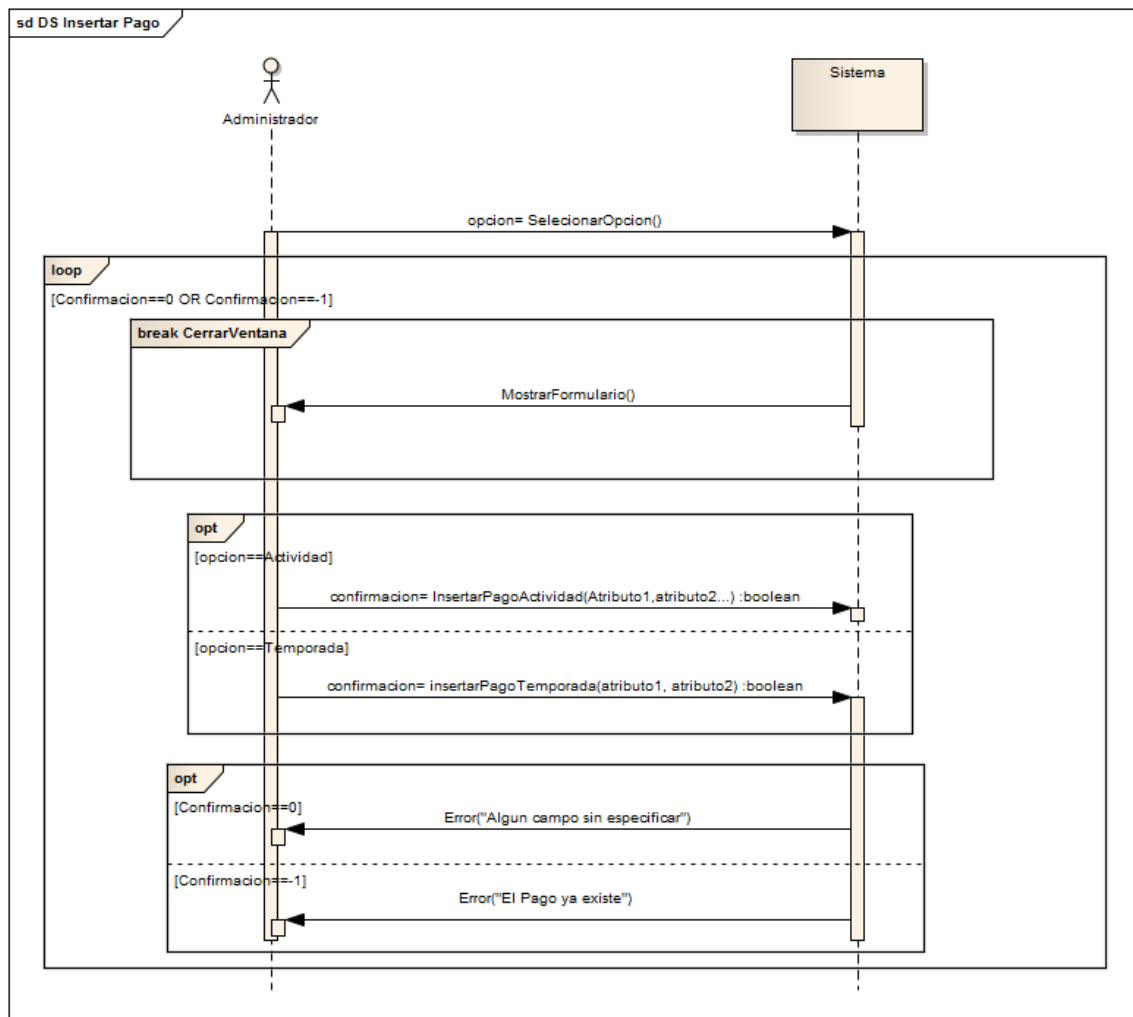


Diagrama de secuencia: Consultar Pago

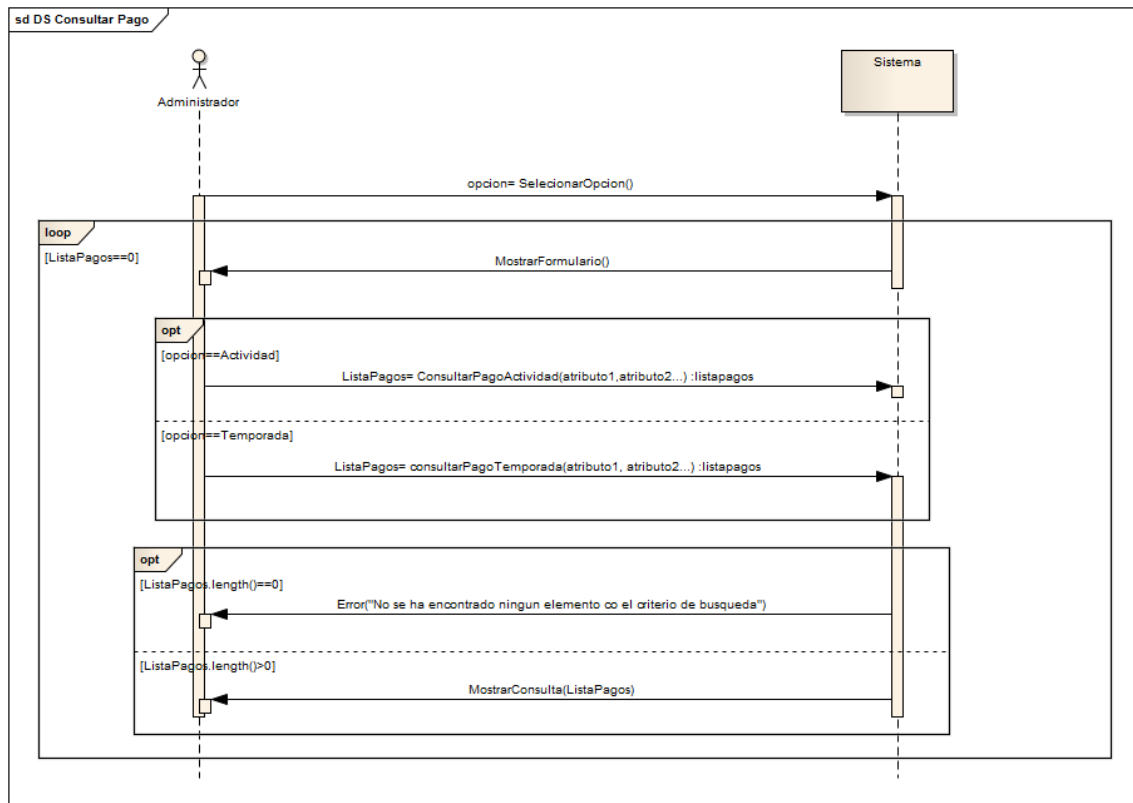


Diagrama de secuencia: Eliminar Pago

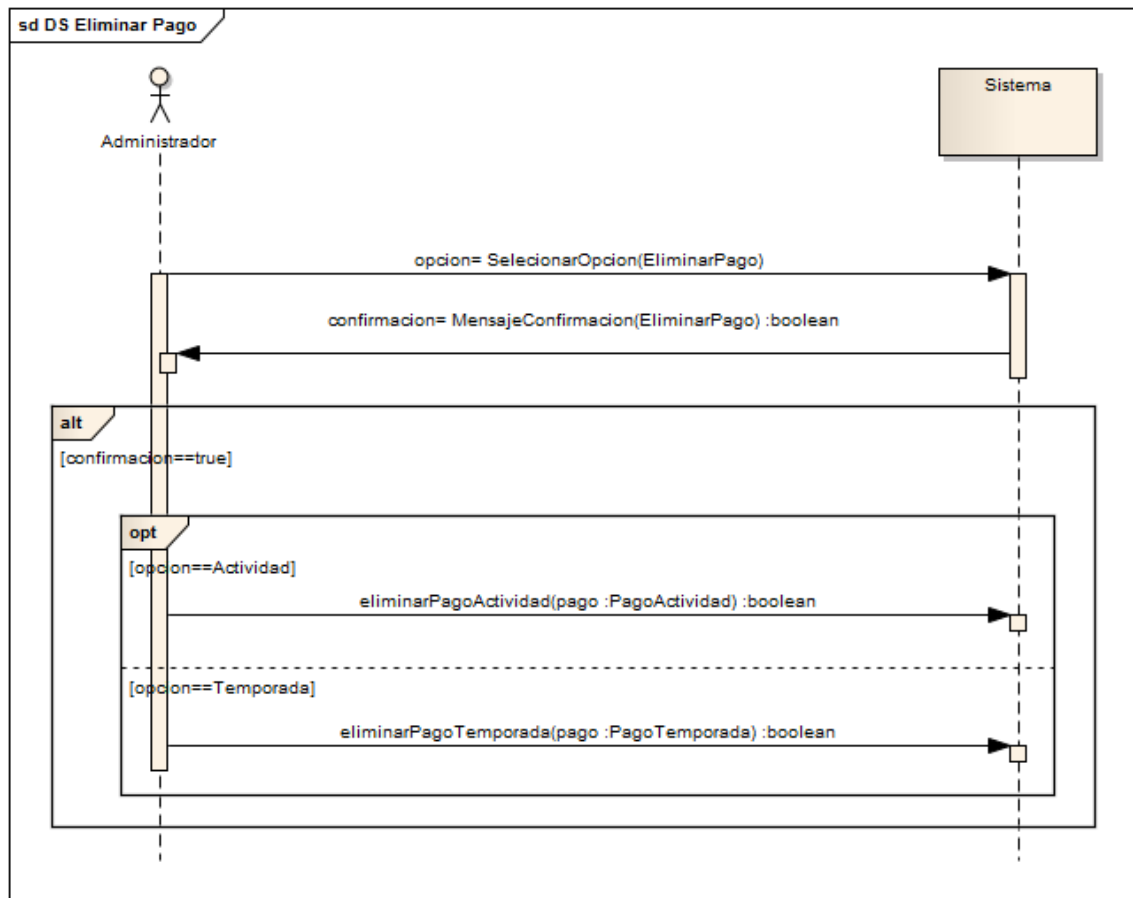
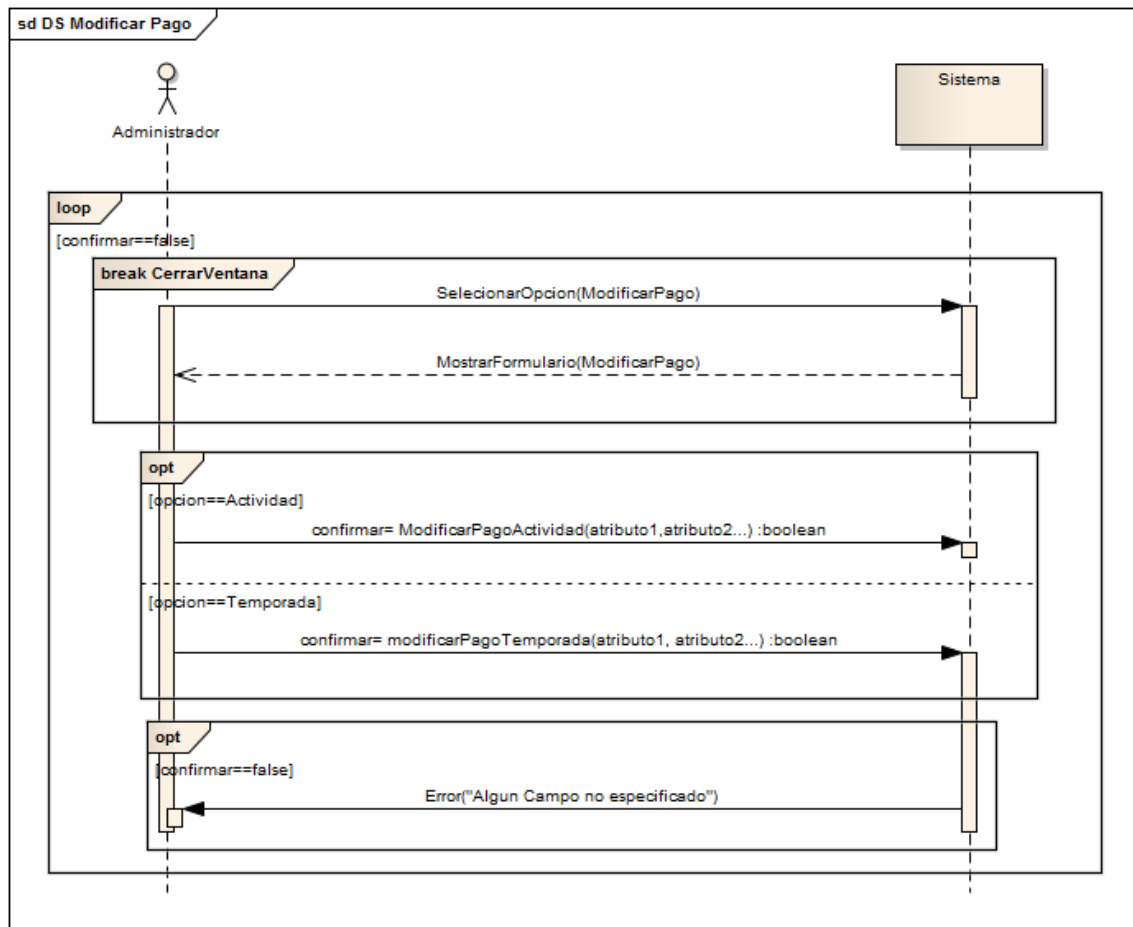


Diagrama de secuencia: Modificar Pago



Gestión de Temporadas

Diagrama de secuencia: Insertar Temporada

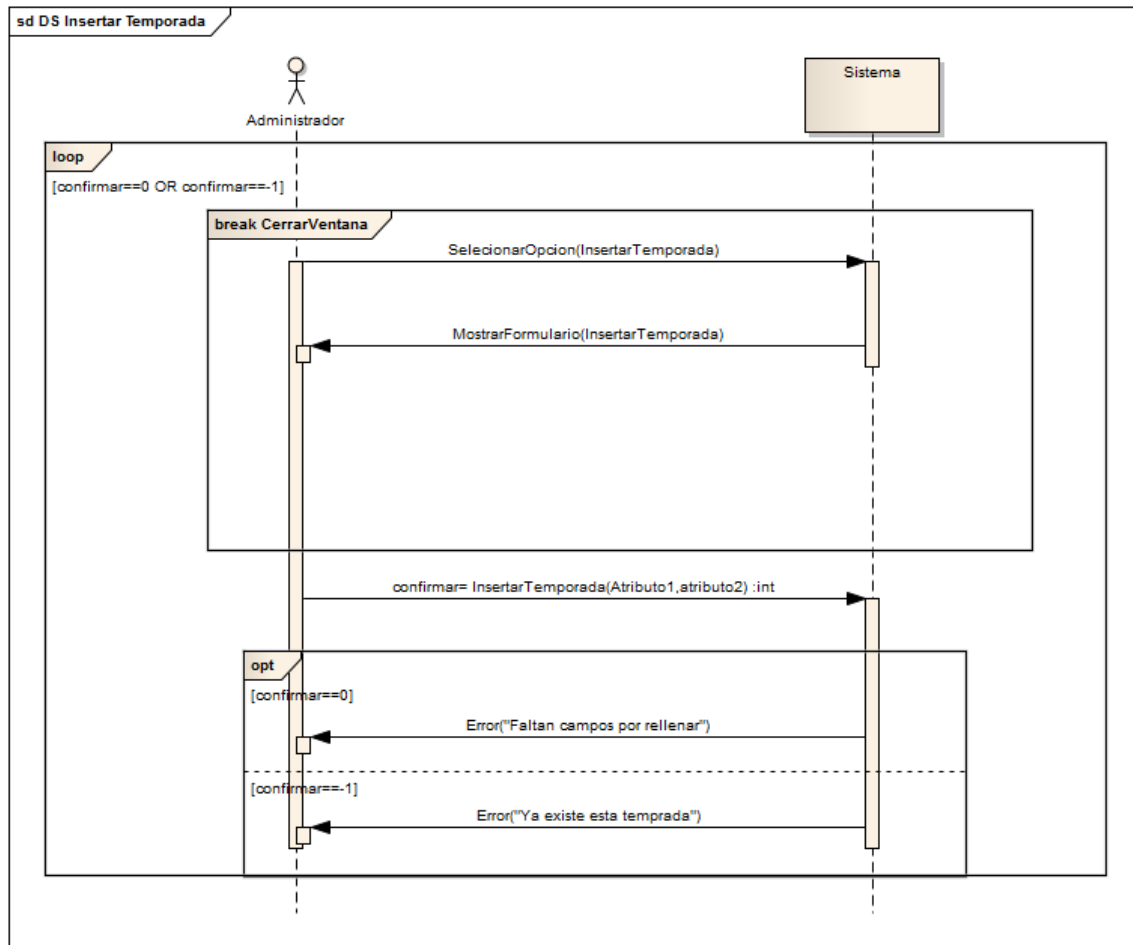


Diagrama de secuencia: Consultar Temporada

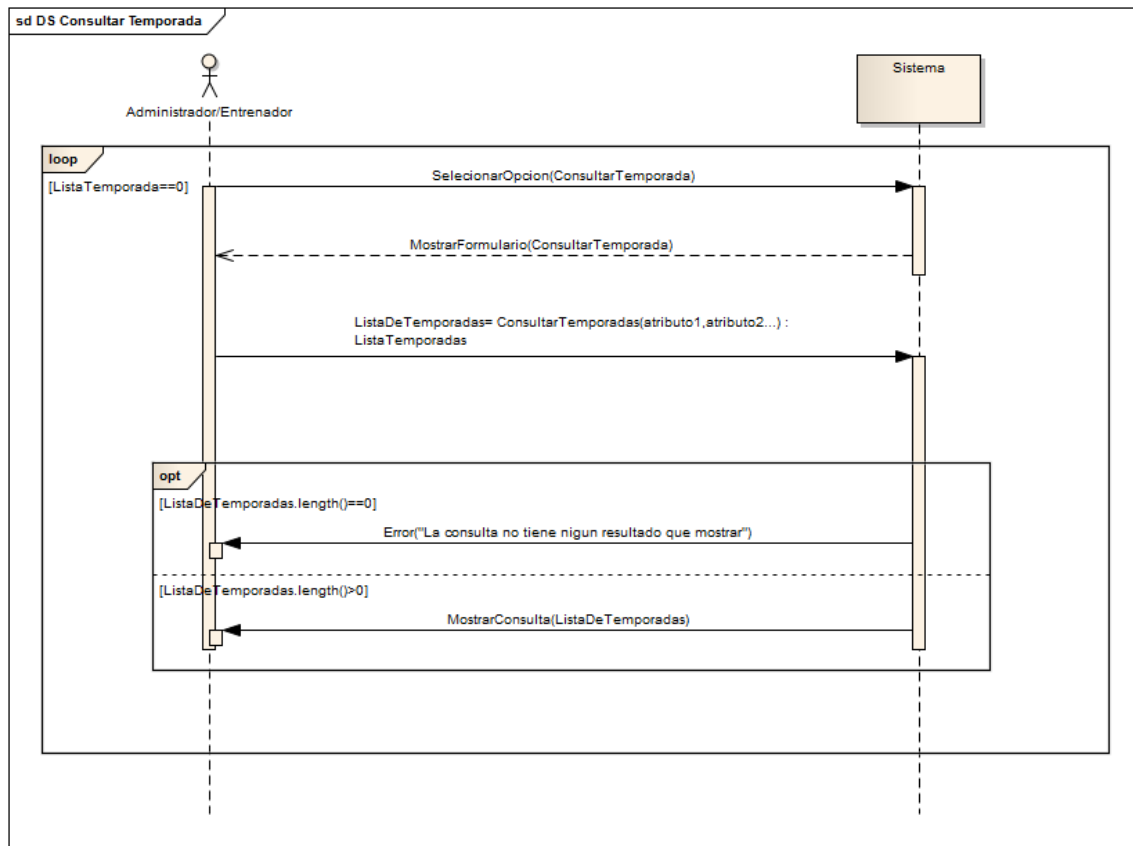


Diagrama de secuencia: Eliminar Temporada

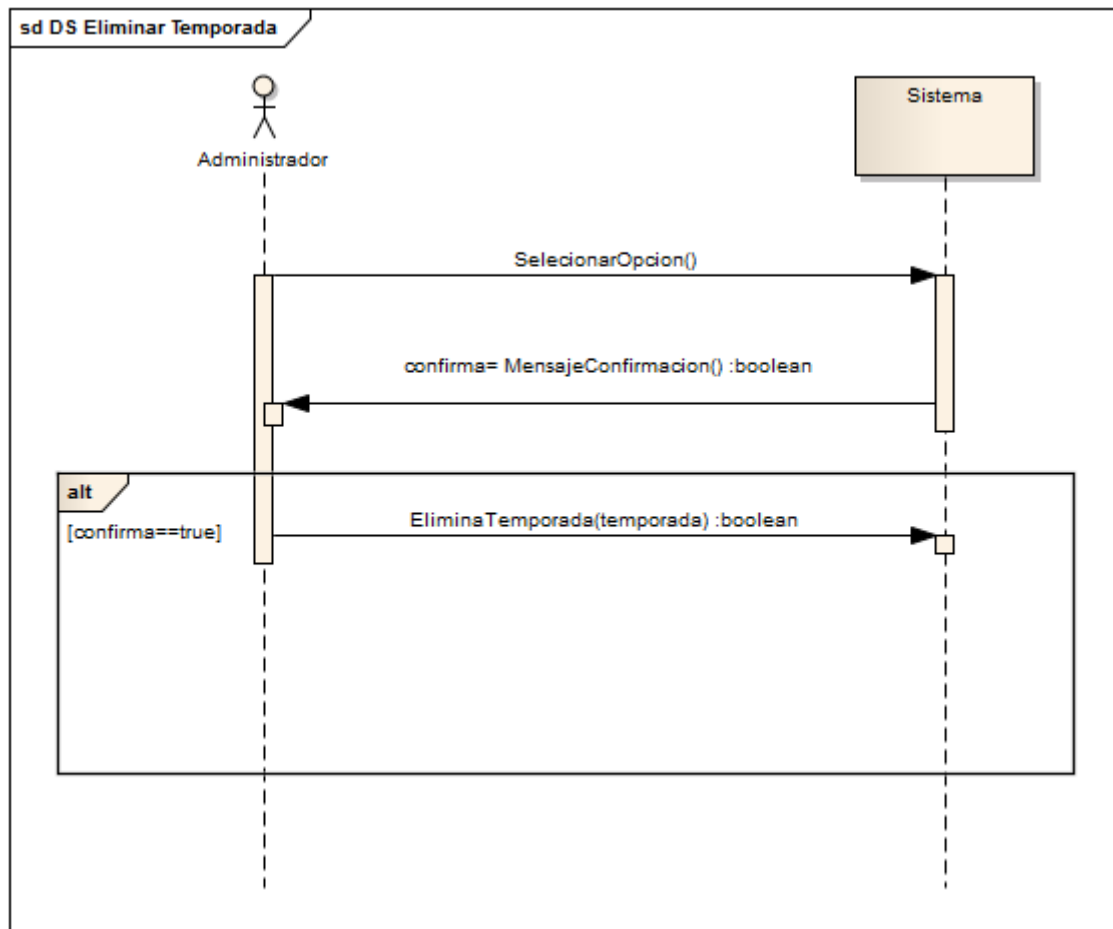
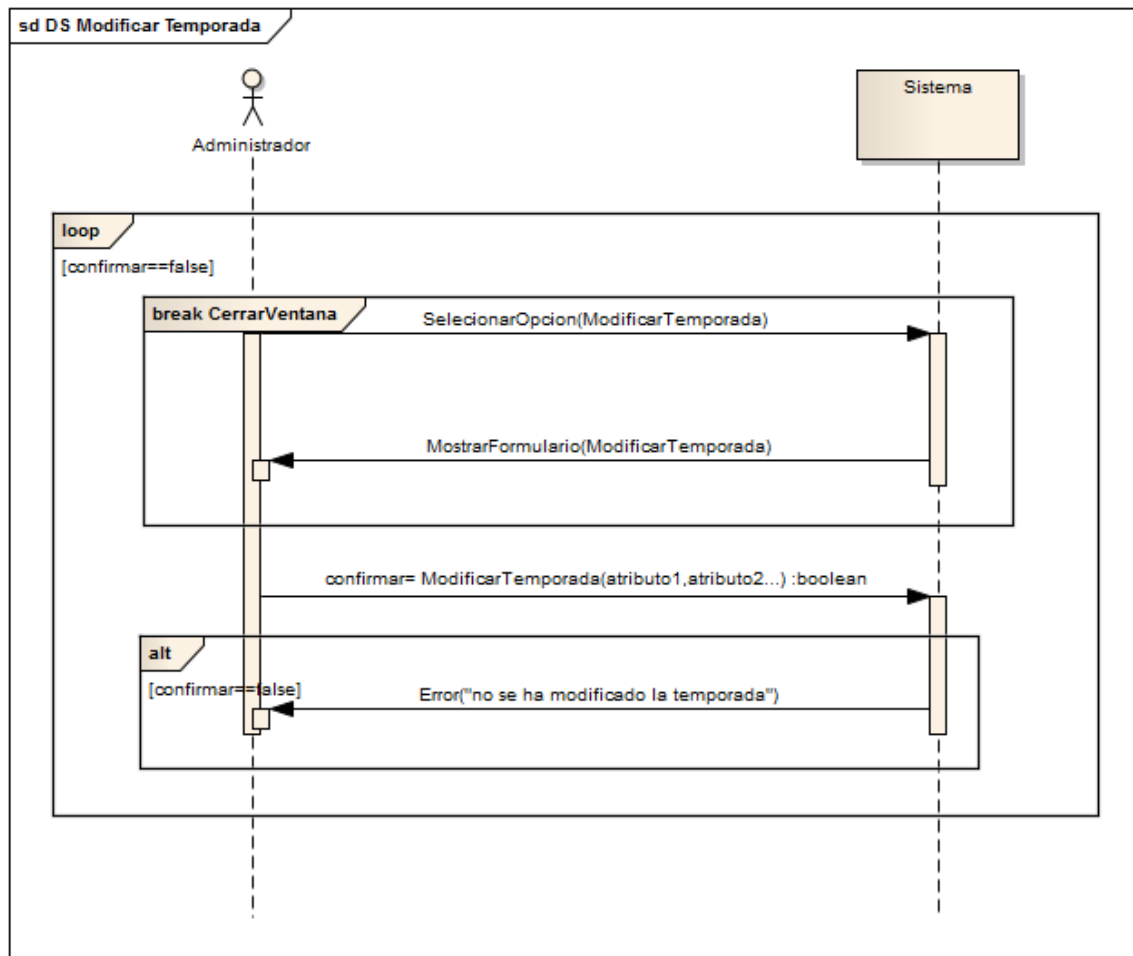


Diagrama de secuencia: Modificar Temporada



Análisis

Identificar clases, atributos y relaciones

En este punto se buscan las abstracciones más significativas que identifiquen los aspectos claves del dominio del problema. Las clases conceptuales deben interrelacionarse entre sí a través de las relaciones para satisfacer las necesidades de información o de comportamiento que demandan los casos de uso y así comprender mejor el modelo.

A continuación se detallan las clases, atributos y relaciones entre las clases que se han obtenido para esta segunda iteración del problema planteado.

Clase: Actividad

Atributos:

- descripcion (string): Descripción breve de la actividad para consultas por el sistema.
- fechaFin (date): Fecha fin de la actividad para la temporada.
- fechaInicio (date): Fecha inicio de la actividad para la temporada.
- nombre (string): Nombre identificativo de la actividad.
- precioNoSocio (float): Es el precio que tendrán que abonar los alumnos que no sean socios y que deseen participar en la actividad.
- precioSocio (float): Es el precio que tendrán que abonar los alumnos que si sean socios y que deseen participar en la actividad.
- idActividad (int): Número identificador de la actividad.

Relaciones:

- Asociación con la clase Alumno.
- Asociación con la clase Temporada.

Clase: Categoría

Atributos:

- descripción (String): Breve descripción de la categoría del alumno.
- edadMaxima (int): La edad máxima posible a la que se puede acceder a la categoría.
- idCategoría (int): Número identificador de la categoría.
- tipo (String): Nombre de la categoría del alumno.

Relaciones:

- Asociación con la clase Equipo.
- Asociación con la clase Grupo.

Clase: Equipo

Atributos:

- idEquipo (int): Número identificador del equipo.
- sexo(Sexo): Este campo corresponde con el sexo que tendrán los integrantes del equipo, que pertenece al tipo enumerado Sexo en donde se establecen todos los valores posibles que puede tomar el campo.

Relaciones:

- Asociación con la clase Categoría.
- Asociación con la clase Entrenador.
- Asociación con la clase Fundación.
- Asociación con la clase Alumno.
- Asociación con la clase Temporada.

Clase: Grupo Entrenamiento

Atributos:

- idGrupo (int): Número identificador del alumno.
- Sexo(Sexo): Este campo corresponde con el sexo que tendrán los integrantes del equipo, que pertenece al tipo enumerado Sexo en donde se establecen todos los valores posibles que puede tomar el campo.

Relaciones:

- Asociación con la clase Entrenador.
- Asociación con la clase Categoría.
- Asociación con la clase Alumno.
- Asociación con la clase Temporada.

Clase: Pago Temporada (clase de asociación entre las clases Temporada y Alumno)

Atributos:

- idPagoTemporada (int): Número identificador del pago por temporada.

Relaciones:

- Asociación con la clase Cuota precio (composición de esta última)

Clase: Pago Actividad (clase de asociación entre Alumno y Actividad)

Atributos:

- idPagoActividad (int): Número identificador del pago de la actividad.
- recibo (string): Ruta relativa o completa del archivo que contiene la imagen del recibo/pago/domiciliación del pago de la actividad.

Relaciones:

- Asociación con la clase Cuota precio (composición).

Clase: Cuota Precio

Atributos:

- idCuotaPrecio (int): Número identificador del precio de la cuota.
- importe (float): Valor numérico del importe de la mensualidad del alumno.
- pagado (boolean): Valor booleano que establece si la mensualidad ha sido pagada o no por el alumno. Se modificará una vez se haya recibido del banco el documento con los pagos del mes.
- fechaPago (date): Fecha en la que se realizó el pago.

Relaciones:

- Asociación con la clase Pago Temporada.
- Asociación con la clase Pago Actividad.

Clase: Temporada

Atributos:

- curso (string): año correspondiente al curso de la temporada (ejemplo: 2012-2013 sería 12/13).
- idTemporada (int): Número identificador de la temporada.

Relaciones:

- Asociación con la clase Equipo.
- Asociación con la clase Grupo.
- Asociación con la clase Alumno.
- Asociación con la clase Temporada.

Clase: Instalación

Atributos:

- capacidadEquipos (int): Indica el número de equipos que pueden entrenar a la vez en esa instalación.
- localizacion (String): Indica la calle y zona donde está ubicada la instalación.
- nombre (String): Es el nombre de la instalación.
- idInstalación (int): Número identificador de la instalación.

Relaciones:

- Asociación con la clase Actividad.
- Asociación con la clase GrupoEntrenamiento.

Clase: Horario (clase de asociación entre Instalación y GrupoEntrenamiento)

Atributos:

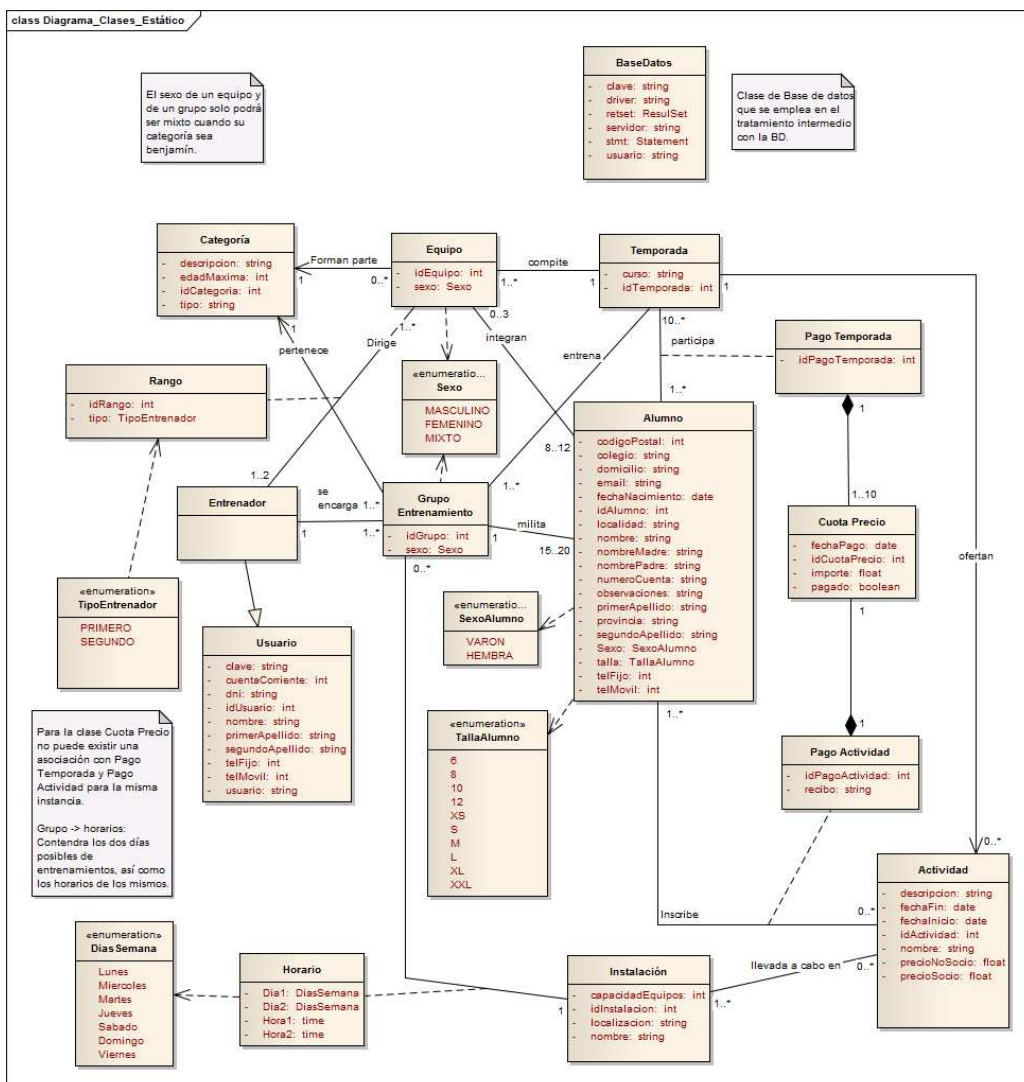
- Dia1 (DiasSemana): Este campo corresponde con un día de la semana en el que el grupo entrenará en la instalación, que pertenece al tipo enumerado DiasSemana en donde se establecen todos los valores posibles que puede tomar el campo.
- Hora1 (time): Este campo indica la hora en la que el grupo entrenará en la instalación el día1.
- Dia2 (DiasSemana): Este campo corresponde con el otro día de la semana en el que el grupo entrenará en la instalación, que pertenece al tipo enumerado DiasSemana en donde se establecen todos los valores posibles que puede tomar el campo.
- Hora2 (time): Este campo indica la hora en la que el grupo entrenará en la instalación el día2.

Modelado estático – Diagrama de clases

El modelo estático proporciona mecanismos para describir y representar las interrelaciones estructurales que se establecen entre las clases conceptuales de un modo visual y compacto.

- Proporciona una estructura estática de las clases conceptuales.
- Al ser un modelo de análisis sólo muestra las clases conceptuales extraídas del dominio de aplicación (dominio del problema) y sus relaciones, no componentes software pertenecientes a la solución.
- La reducida barrera que propugna la orientación a objetos entre el problema y la solución, posibilita que el modelo estático inspire la construcción del modelo de diseño.
- Definen un vocabulario común entre clientes y desarrolladores.
- Una vez que el modelo es estable, la descripción de cada clase será tan detallada como sea posible.
- Elimina la vaguedad en la definición de las clases del dominio del problema.

A continuación se describe el diagrama de clases obtenido para el problema especificado.



Modelado del comportamiento externo – Contratos

Describen el comportamiento detallado del sistema en función de los cambios de estado de los objetos cuando se invoca una operación del sistema.

Es un documento que describe lo que una operación se propone lograr, sin decir cómo se conseguirá.

- Define la especificación de una operación sin entrar en su implementación.
- Suele redactarse con un estilo declarativo.

Contratos de los casos de uso de la Gestión de Actividades

Contratos de Caso de uso: Insertar Actividad

Nombre	InsertarAlumnos(listaAlumnos: lista)
Responsabilidades	Introduce los alumnos que van a disfrutar de la actividad.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Ninguno de los campos puede ser nulo.
Postcondición	Los alumnos se han insertado en la lista.

Nombre	InsertarInstalacion(listaInstalaciones: lista)
Responsabilidades	Introduce las instalaciones donde va a desarrollarse la actividad.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Ninguno de los campos puede ser nulo.
Postcondición	Las instalaciones se han insertado en la lista.

Nombre	InsertarDatosActividad(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos de la actividad que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	A más a más de los atributos de un grupo se le pasará la lista de añadidos por parámetro para realizar los cambios que sean necesarios en las listas de alumnos inscritos.
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	La actividad ha sido creada e inicializada correctamente con todos sus atributos, y alumnos e instalaciones, en el caso de que se hayan especificado.

Contratos de Caso de uso: Consultar Actividad

Nombre	consultarActividad(atributo1, atributo 2...)
Responsabilidades	Consultar varias actividades.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todas las actividades que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de actividades no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Actividad

Nombre	eliminaActividad(actividad: Actividad)
Responsabilidades	Elimina una actividad existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente la actividad que deseamos eliminar.
Postcondición	La actividad se ha borrado totalmente del sistema.

Contratos de Caso de uso: Modificar Actividad

La función de InsertarAlumnos no necesita contrato ya que ha sido definida en el contrato que lleva el mismo nombre más arriba, en los contratos del caso de uso Insertar Actividad.

Nombre	eliminarAlumnos(ListaAlumnos: lista)
Responsabilidades	Eliminalos alumnosde una lista de una actividad existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente la actividad que deseamos modificar y que el alumno esté dado de alta en el sistema.
Postcondición	El alumno ha sido añadido a la lista de eliminados.

Nombre	modificarActividad(atributo1, atributo2...)
Responsabilidades	Modificauna actividad existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	A más a más de los atributos de un grupo se le pasaran las dos listas de añadidos y eliminados por parámetro para realizar los cambios que sean necesarios en las listas de alumnos inscritos.
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente la actividad que deseamos modificar.
Postcondición	El sistema modifica los datos de la actividad.

Contratos de los casos de uso de la Gestión de Categorías

Contratos de Caso de uso: InsertarCategoría

Nombre	InsertarCategoría(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos de la categoría que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	La categoría ha sido creada e inicializada correctamente con todos sus atributos.

Contratos de Caso de uso: Consultar Categoría

Nombre	consultarCategoría(atributo1, atributo 2...)
Responsabilidades	Consultar varias categorías.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todas las categorías que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de categorías no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Categoría

Nombre	eliminarCategoría(categoría: Categoría)
Responsabilidades	Elimina una categoría existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente la categoría que deseamos eliminar.
Postcondición	La categoría se ha borrado totalmente del sistema.

Contratos de Caso de uso: Modificar Categoría

Nombre	modificarCategoría(atributo1, atributo2...)
Responsabilidades	Modifica una categoría existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente la categoría que deseamos modificar.
Postcondición	El sistema modifica los datos de la categoría.

Contratos de los casos de uso de la Gestión de Equipos

Contratos de Caso de uso: Insertar Equipo

Nombre	InsertarAlumnosEquipo(listaAlumnos: lista)
Responsabilidades	Introduce los alumnos que van a formar parte del equipo.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Ninguno de los campos puede ser nulo.
Postcondición	Los alumnos se han insertado en la lista de añadidos.

Nombre	InsertarEquipo(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos del equipo que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	A más a más de los atributos de un grupo se le pasará la lista de añadidos por parámetro para realizar los cambios que sean necesarios en las listas de alumnos inscritos.
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	El equipo ha sido creado e inicializado correctamente con todos sus atributos, y alumnos, en el caso de que se hayan especificado.

Contratos de Caso de uso: Consultar Equipo

Nombre	consultarEquipo(atributo1, atributo 2...)
Responsabilidades	Consultar varios equipos.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todos los equipos que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de equipos no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Equipo

Nombre	eliminarEquipo(equipo: Equipo)
Responsabilidades	Elimina un equipo existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente el equipo que deseamos eliminar.
Postcondición	El equipo se ha borrado totalmente del sistema.

Contratos de Caso de uso: Modificar Equipo

La función de InsertarAlumnosEquipo no necesita contrato ya que ha sido definida en el contrato que lleva el mismo nombre más arriba, en los contratos del caso de uso Insertar Equipo.

Nombre	eliminarAlumnoEquipo(ListaAlumnos: lista)
Responsabilidades	Elimina los alumnos de una lista de unequipoexistente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente elequipo que deseamos modificar y que el alumno esté dado de alta en el sistema.
Postcondición	El alumno ha sido añadido a la lista de eliminados.

Nombre	modificarEquipo(atributo1, atributo2...)
Responsabilidades	Modifica unequipoexistente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	A más a más de los atributos de un grupo se le pasaran las dos listas de añadidos y eliminados por parámetro para realizar los cambios que sean necesarios en las listas de alumnos inscritos.
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente elequipo que deseamos modificar.
Postcondición	El sistema modifica los datos delequipo.

Contratos de los casos de uso de la Gestión de Grupos

Contratos de Caso de uso: Insertar Grupo

Nombre	InsertarAlumnosGrupo(listaAlumnos: lista)
Responsabilidades	Introduce los alumnos que van a formar parte del grupo.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Ninguno de los campos puede ser nulo.
Postcondición	Los alumnos se han insertado en la lista de añadidos.

Nombre	InsertarGrupo(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos del grupo que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	A más a más de los atributos de un grupo se le pasará la lista de añadidos por parámetro para realizar los cambios que sean necesarios en las listas de alumnos inscritos.
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	El grupo ha sido creado e inicializado correctamente con todos sus atributos, y alumnos, en el caso de que se hayan especificado.

Contratos de Caso de uso: Consultar Grupo

Nombre	consultarGrupo(atributo1, atributo 2...)
Responsabilidades	Consultar varios grupos.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todos los grupos que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de grupos no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Grupo

Nombre	eliminarGrupo(grupo: Grupos)
Responsabilidades	Elimina un grupo existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente el grupo que deseamos eliminar.
Postcondición	El grupo se ha borrado totalmente del sistema.

Contratos de Caso de uso: *Modificar Grupo*

La función de InsertarAlumnosEquipo no necesita contrato ya que ha sido definida en el contrato que lleva el mismo nombre más arriba, en los contratos del caso de uso Insertar Equipo.

Nombre	eliminarAlumnosGrupo(ListaAlumnos: lista)
Responsabilidades	Elimina los alumnos de una lista de un grupo existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente el grupo que deseamos modificar y que el alumno esté dado de alta en el sistema.
Postcondición	El alumno ha sido añadido a la lista de eliminados.

Nombre	modificarGrupo(atributo1, atributo2...)
Responsabilidades	Modifica un grupo existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	A más a más de los atributos de un grupo se le pasaran las dos listas de añadidos y eliminados por parámetro para realizar los cambios que sean necesarios en las listas de alumnos inscritos.
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente el grupo que deseamos modificar.
Postcondición	El sistema modifica los datos del grupo.

Contratos de los casos de uso de la Gestión de Instalaciones

Contratos de Caso de uso: Insertar Instalación

Nombre	InsertarInstalacion(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos de la instalación que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	La instalación ha sido creada e inicializada correctamente con todos sus atributos.

Contratos de Caso de uso: Consultar Instalación

Nombre	consultarInstalacion(atributo1, atributo 2...)
Responsabilidades	Consultar varias instalaciones.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todas las instalaciones que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de instalaciones no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Instalación

Nombre	eliminarInstalacion(instalacion: Instalacion)
Responsabilidades	Elimina una instalación existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente la instalación que deseamos eliminar.
Postcondición	La instalación se ha borrado totalmente del sistema.

Contratos de Caso de uso: Modificar Instalación

Nombre	modificarInstalacion(atributo1, atributo2...)
Responsabilidades	Modifica una instalación existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente la instalación que deseamos modificar.
Postcondición	El sistema modifica los datos de la instalación.

Contratos de los casos de uso de la Gestión de Pagos

Contratos de Caso de uso: Insertar Pago

Nombre	InsertarPagoActividad(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos del pago de una actividad que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	El pago ha sido creado e inicializado correctamente con todos sus atributos.

Nombre	InsertarPagoTemporada(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos del pago de la temporada que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	El pago ha sido creado e inicializado correctamente con todos sus atributos.

Contratos de Caso de uso: Consultar Pago

Nombre	consultarPagoActividad(atributo1, atributo 2...)
Responsabilidades	Consultar varios pagos de una actividad.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todos los pagos que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de pagos no deberá ser nula en ningún caso.

Nombre	consultarPagoTemporada(atributo1, atributo 2...)
Responsabilidades	Consultar varios pagos de una temporada.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todos los pagos que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de pagos no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Pago

Nombre	eliminarPagoActividad(pago: PagoActividad)
Responsabilidades	Elimina un pago de una actividad existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente el pago que deseamos eliminar.
Postcondición	El pago se ha borrado totalmente del sistema.

Nombre	eliminarPagoTemporada(pago: PagoTemporada)
Responsabilidades	Elimina un pago de una temporada existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente el pago que deseamos eliminar.
Postcondición	El pago se ha borrado totalmente del sistema.

Contratos de Caso de uso: Modificar Pago

Nombre	modificarPagoActividad(atributo1, atributo2...)
Responsabilidades	Modifica un pago de una actividad existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente el pago que deseamos modificar.
Postcondición	El sistema modifica los datos del pago.

Nombre	modificarPagoTemporada(atributo1, atributo2...)
Responsabilidades	Modifica un pago de una temporada existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente el pago que deseamos modificar.
Postcondición	El sistema modifica los datos del pago.

Contratos de los casos de uso de la Gestión de Temporadas

Contratos de Caso de uso: Insertar Temporada

Nombre	InsertarTemporada(atributo1: tipo1, atributo2: tipo2...)
Responsabilidades	Introduce los datos de la temporada que estamos creando en el sistema.
Tipo	Sistema
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Los campos que no pueden ser nulos no deben serlo.
Postcondición	La temporada ha sido creada e inicializada correctamente con todos sus atributos.

Contratos de Caso de uso: Consultar Temporada

Nombre	consultarTemporada(atributo1, atributo 2...)
Responsabilidades	Consultar varias temporadas.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve una lista con todas las temporadas que ha encontrado según los criterios de búsqueda establecidos.
Precondición	-
Postcondición	La lista de temporadas no deberá ser nula en ningún caso.

Contratos de Caso de uso: Eliminar Temporada

Nombre	eliminarTemporada(temporada: Temporada)
Responsabilidades	Elimina una temporada existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	-
Precondición	Que hayamos seleccionado previamente la temporada que deseamos eliminar.
Postcondición	La temporada se ha borrado totalmente del sistema.

Contratos de Caso de uso: Modificar Temporada

Nombre	modificarTemporada(atributo1, atributo2...)
Responsabilidades	Modifica una temporada existente en el sistema.
Tipo	Sistema.
Referencias cruzadas	-
Notas	-
Excepciones	-
Salida	Devuelve si la operación ha sido efectuada correctamente.
Precondición	Que hayamos seleccionado previamente la temporada que deseamos modificar.
Postcondición	El sistema modifica los datos de la temporada.

Diseño

Descomposición del sistema en subsistemas de diseño para obtener la arquitectura del sistema.

Establecer la arquitectura del sistema.

Para llevar a cabo la fase de diseño hay que establecer la arquitectura del sistema teniendo en cuenta los siguientes puntos:

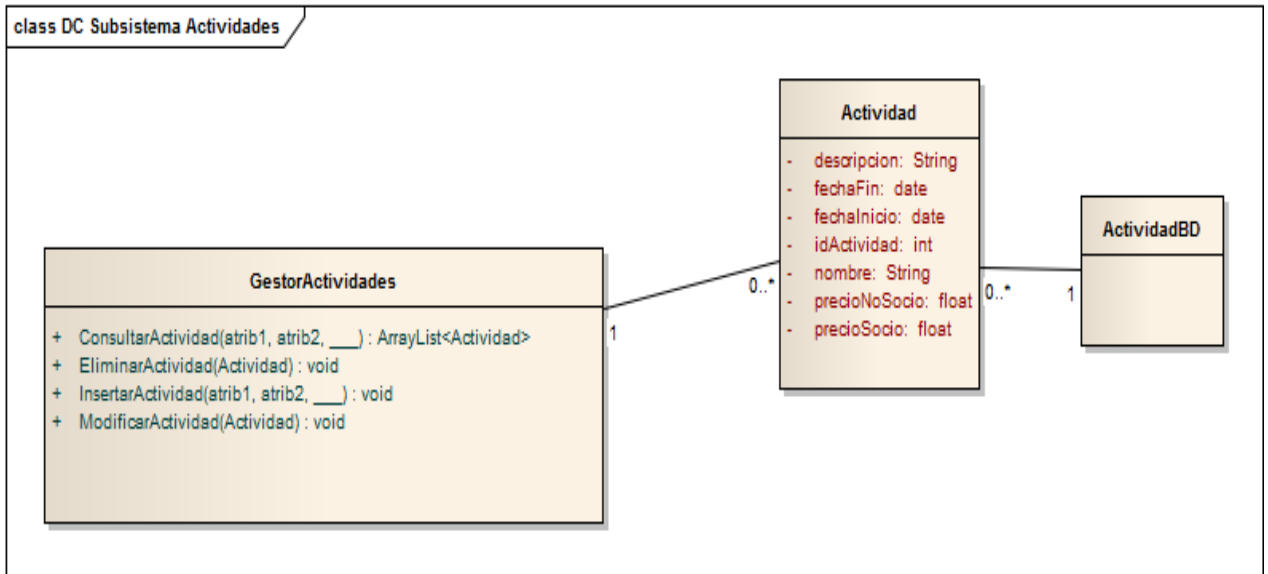
- Objetivos de diseño del sistema.
- Estilos arquitectónicos posibles.
- Documentación de las etapas anteriores del proceso de desarrollo.

Subsistemas funcionales.

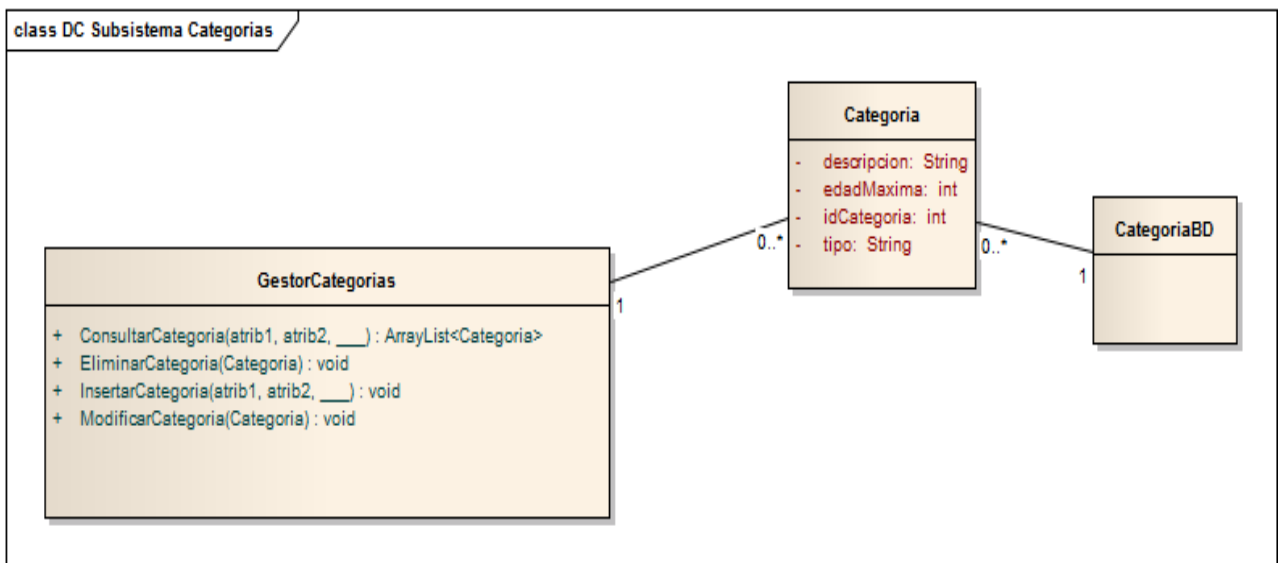
En el modelado de requisitos se han detectado los siguientes subsistemas funcionales:

- Subsistema GestionAlumnos.
- Subsistema GestionUsuarios.
- Subsistema GestiónPagos.
- Subsistema GestiónGruposEntrenamiento.
- Subsistema GestiónEquipos.
- Subsistema GestiónActividades.
- Subsistema GestiónTemporadas.
- Subsistema GestiónCategorías.
- Subsistema GestiónInstalaciones.

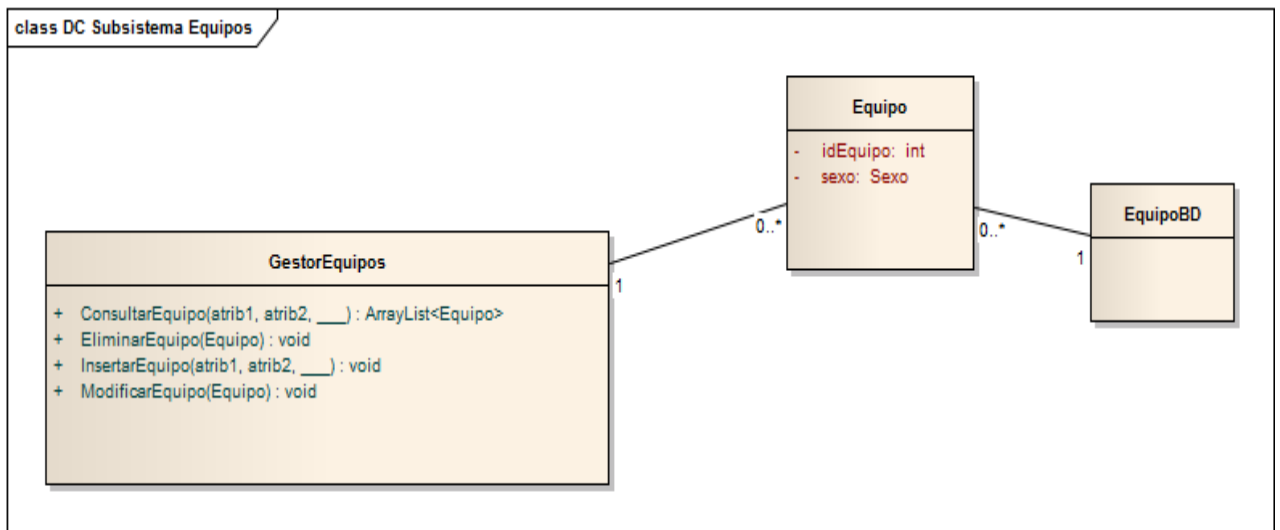
Subsistema Gestión de Actividades



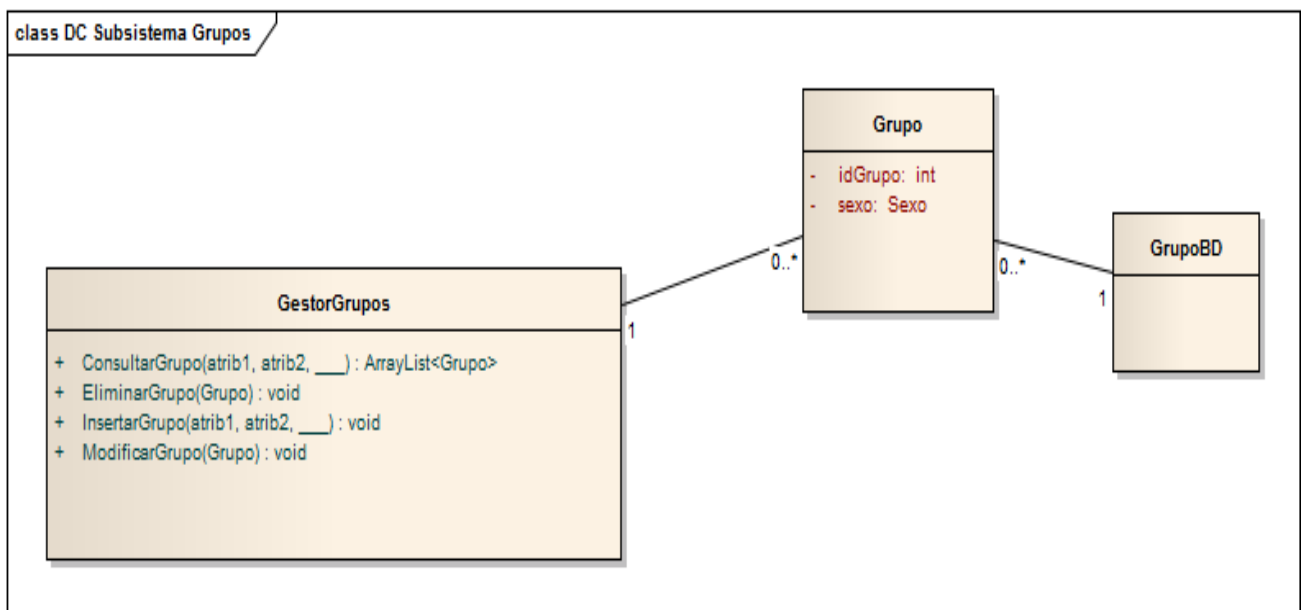
Subsistema Gestión de Categorías



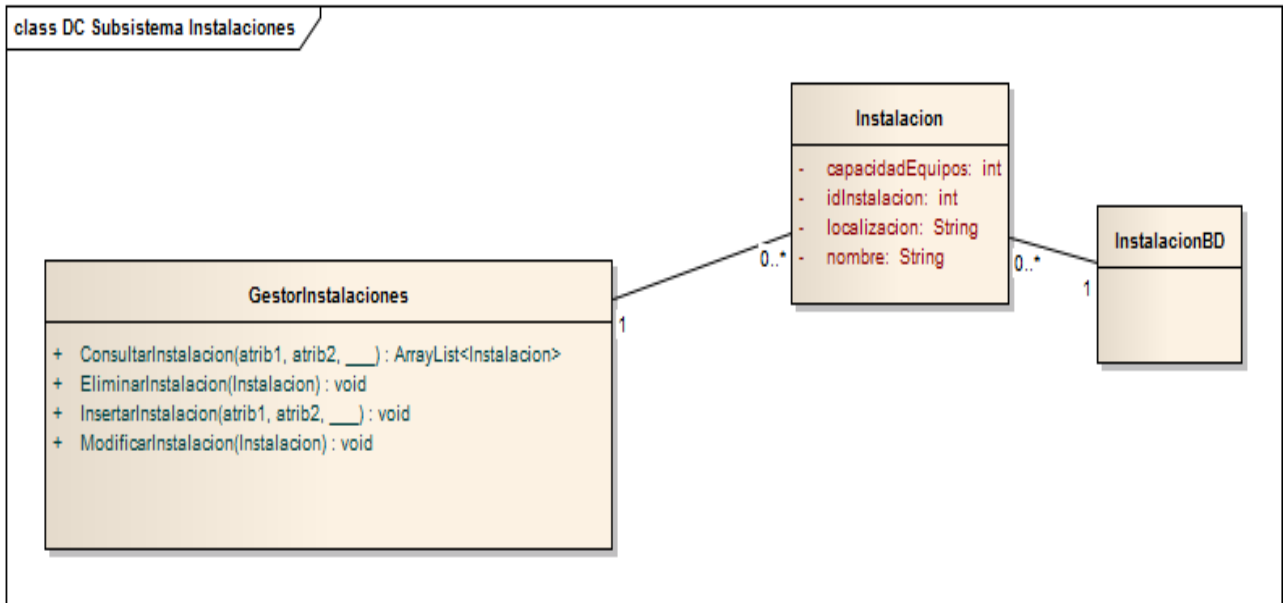
Subsistema Gestión de Equipos



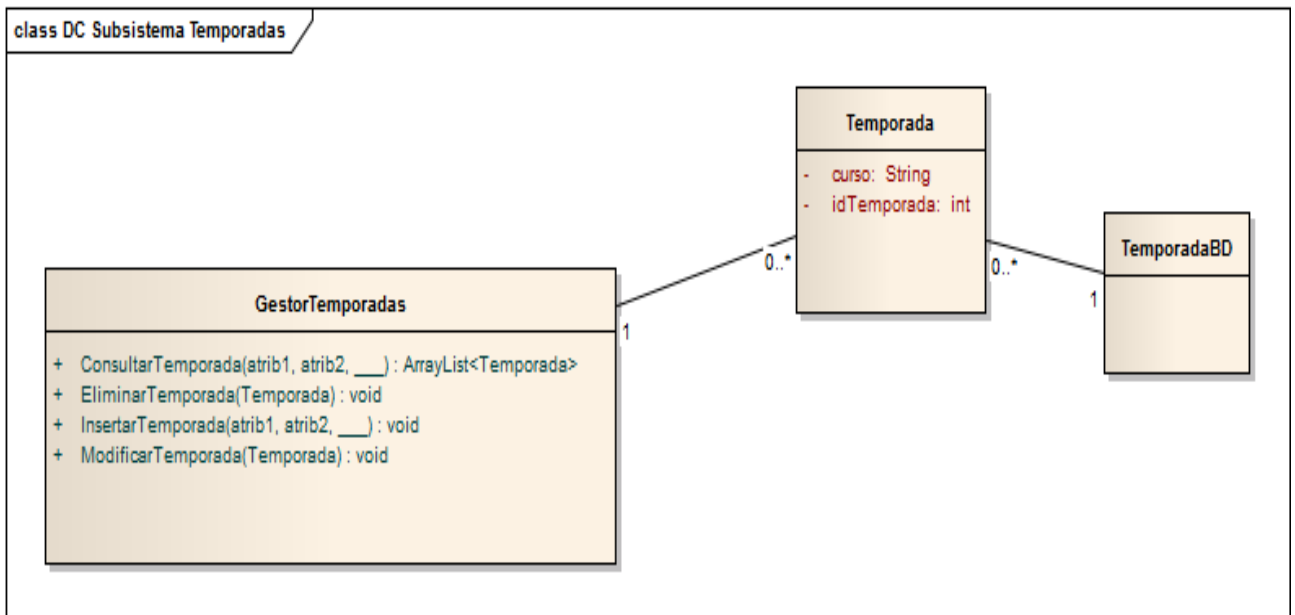
Subsistema Gestión de Grupos



Subsistema Gestión de Instalaciones

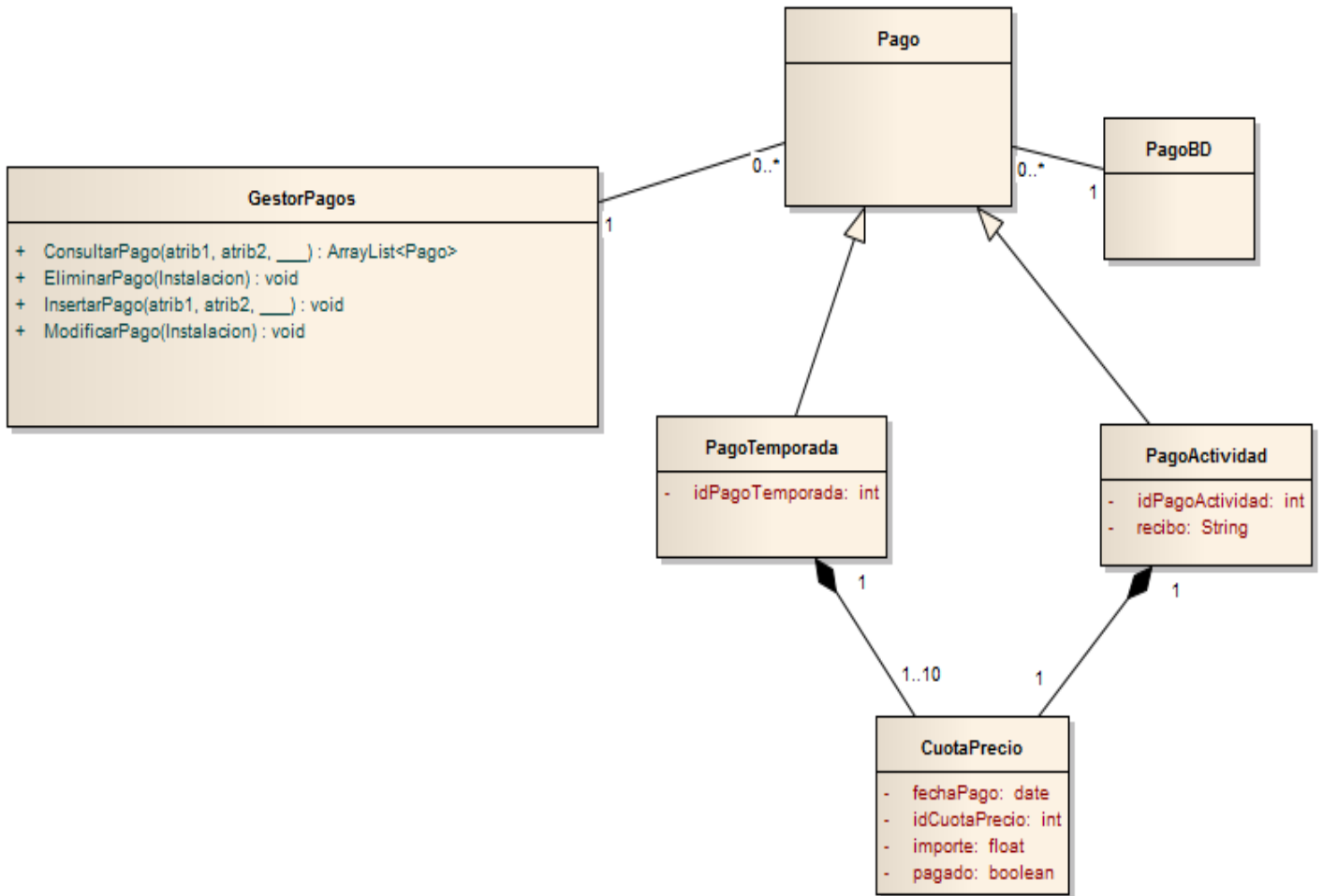


Subsistema Gestión de Temporadas



Subsistema Gestión de Pagos

class DC Subsistema Pagos



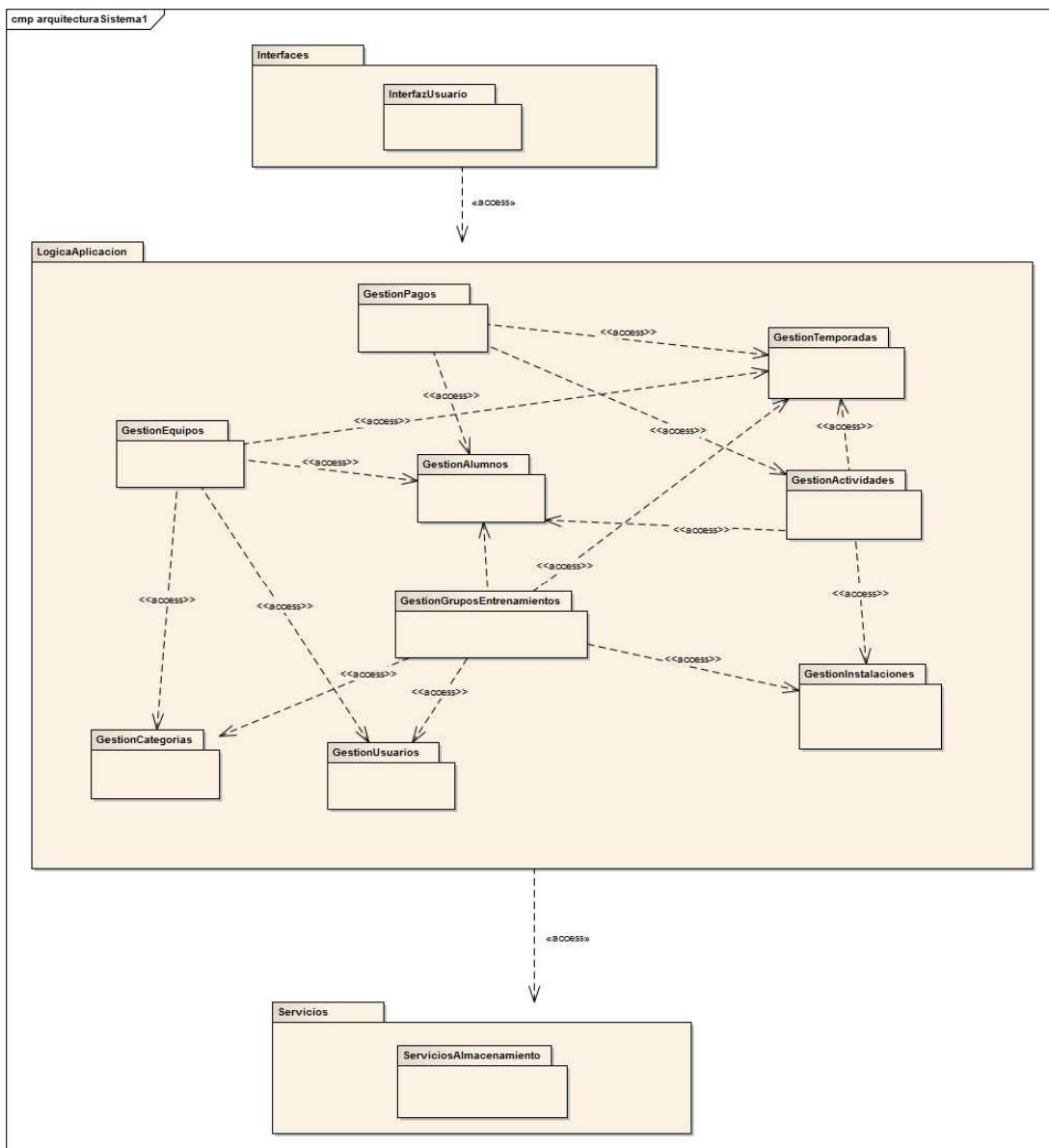
Determinación de la arquitectura.

Después de analizar los objetivos de diseño y los estilos arquitectónicos para el sistema se decide:

Utilizar una arquitectura software basada en capas cerradas que separe la aplicación en tres niveles ya que esto mejora la estructura del sistema y fomenta la flexibilidad en cuanto a sustitución de servicios o interfaz de usuario.

- Interfaz de usuario.
- Lógica de aplicación.
- Servicios

Diagrama arquitectura software de tres capas cerradas



Diagramas de Secuencia de Diseño

Diagrama de secuencia: Consultar Actividad

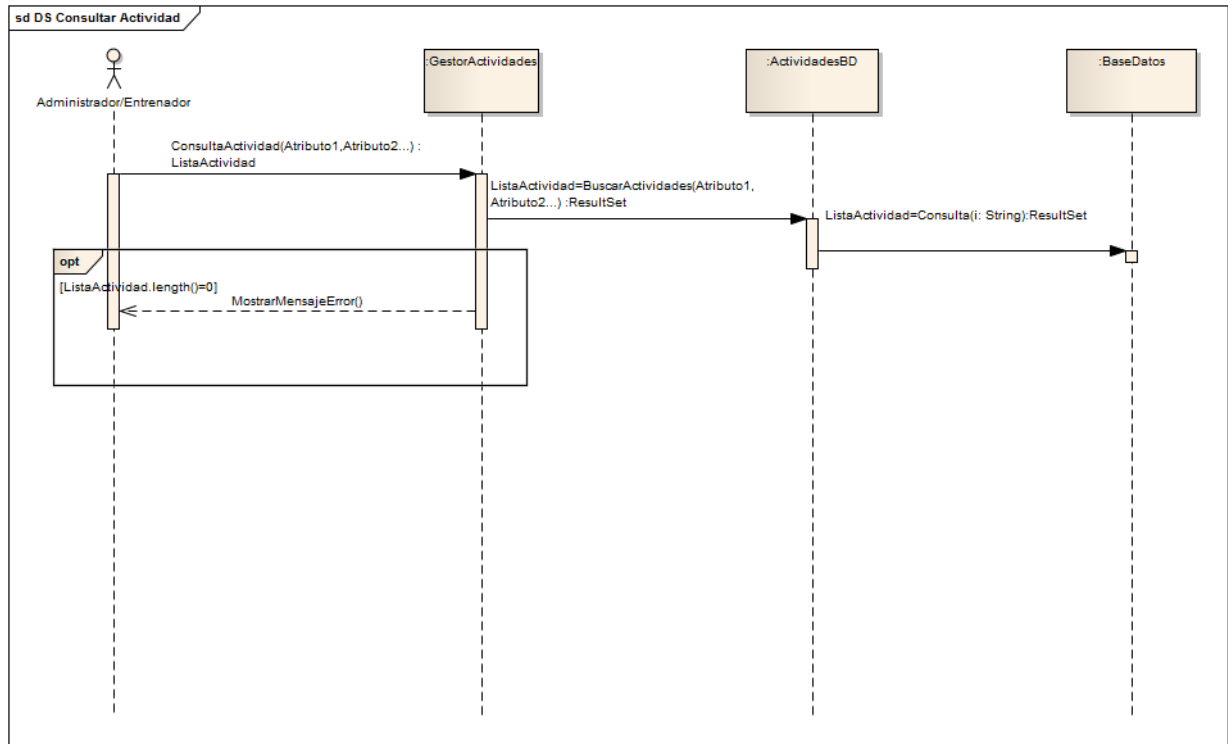


Diagrama de secuencia: Eliminar Actividad

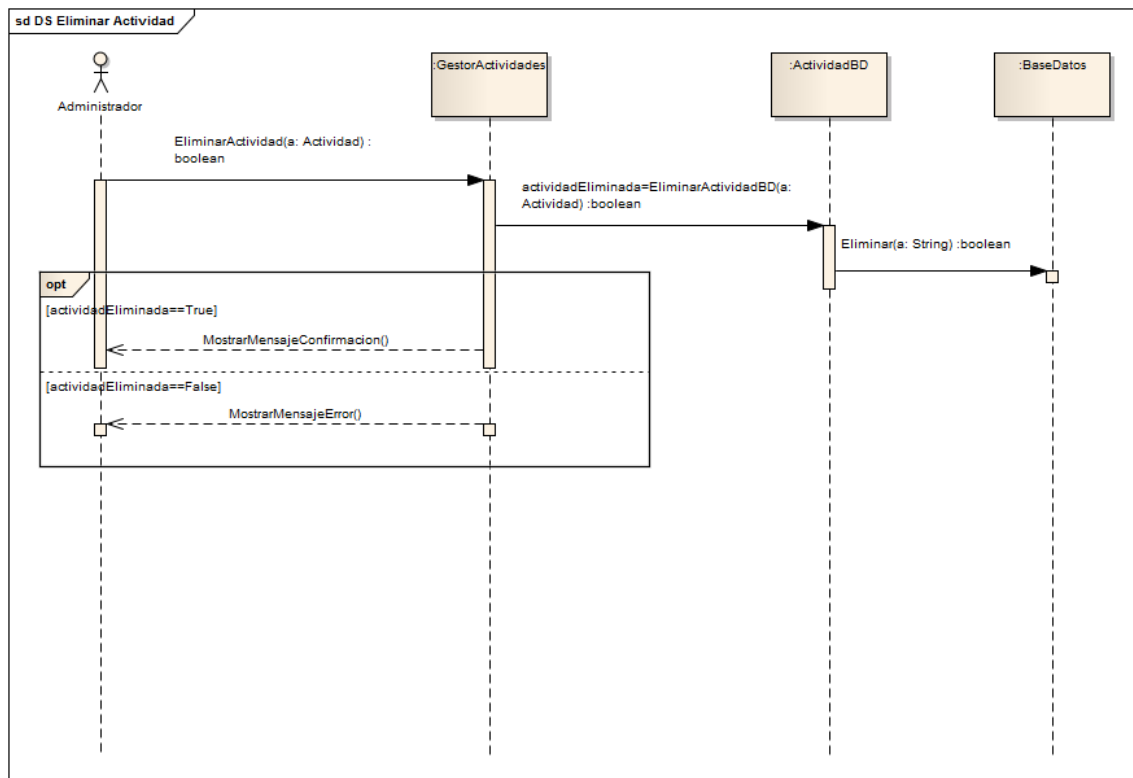


Diagrama de secuencia: Insertar Actividad

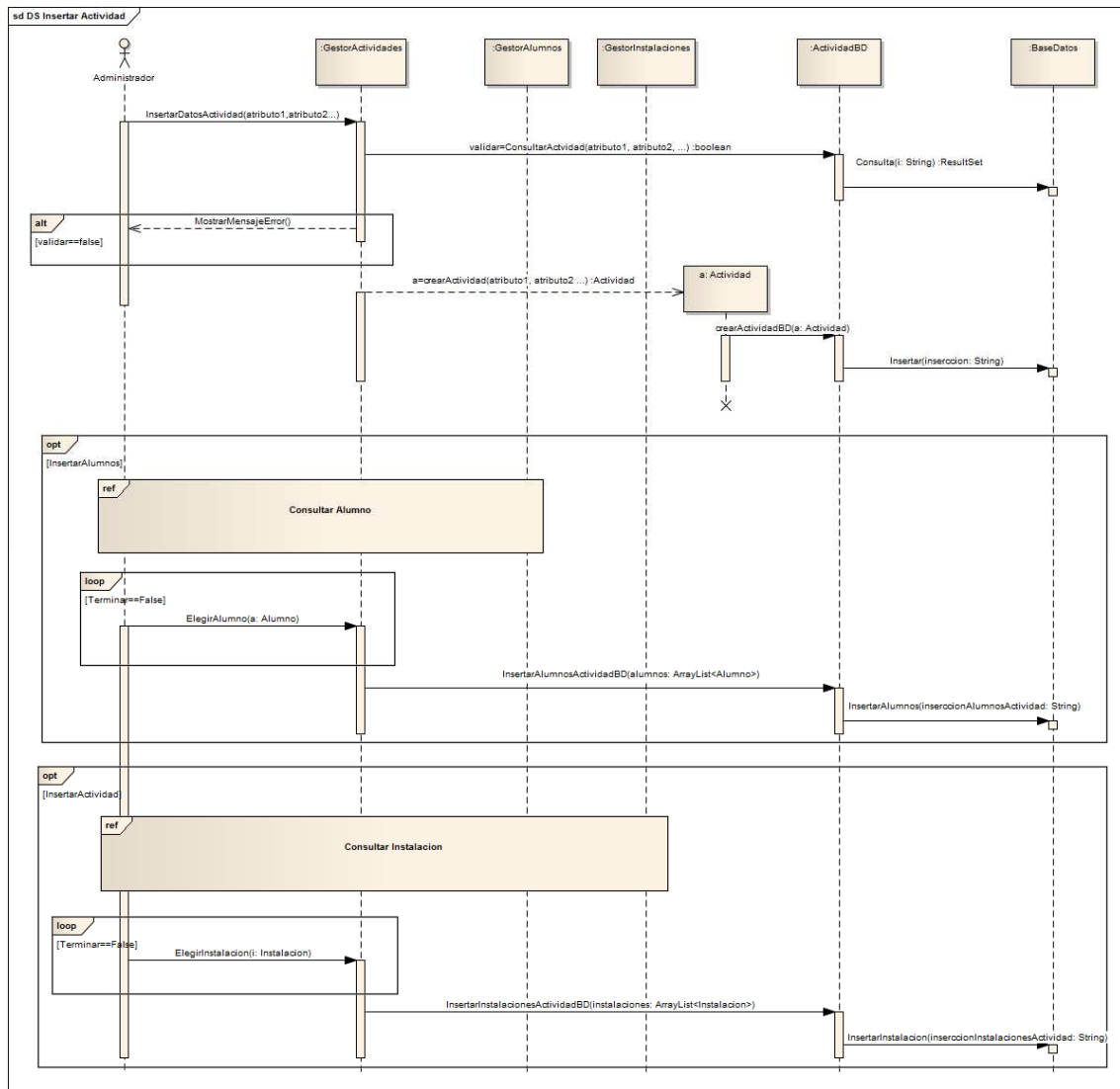


Diagrama de secuencia: Modificar Actividad

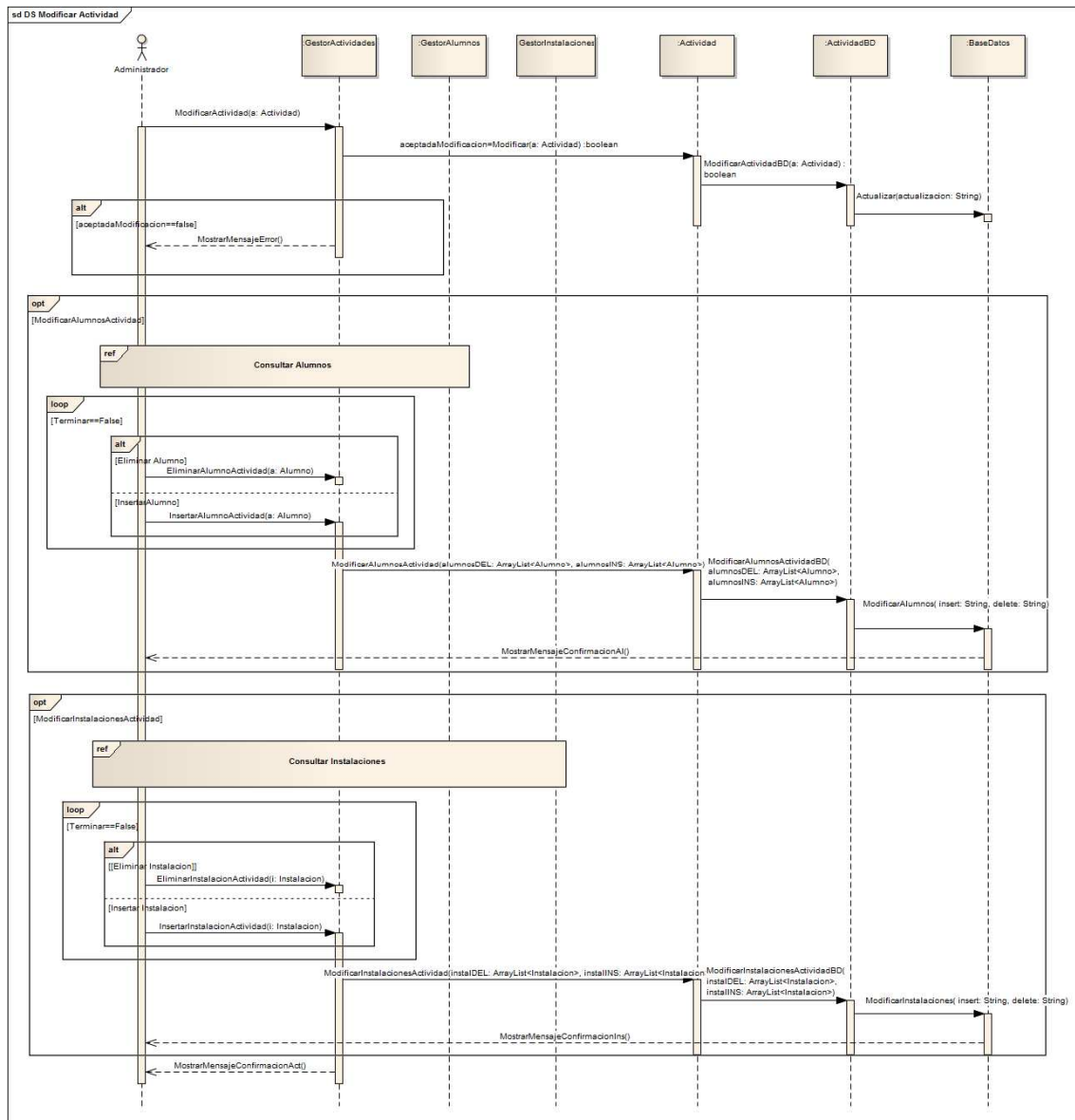


Diagrama de secuencia: ConsultarCategorías

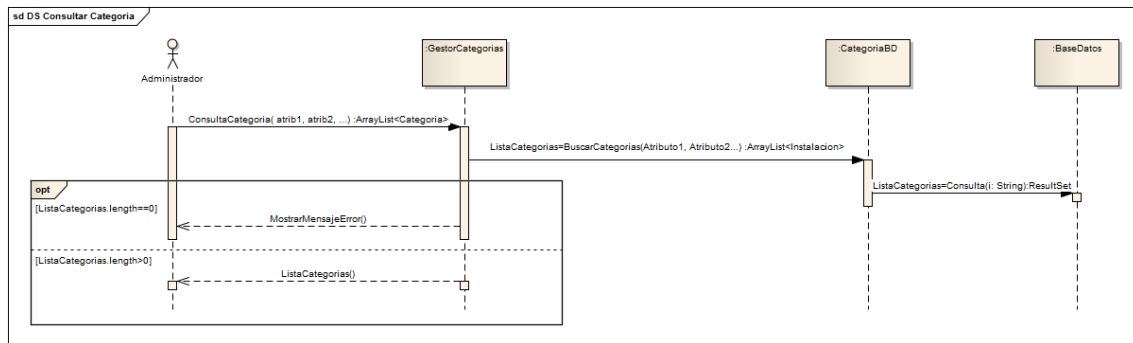


Diagrama de secuencia: EliminarCategorías

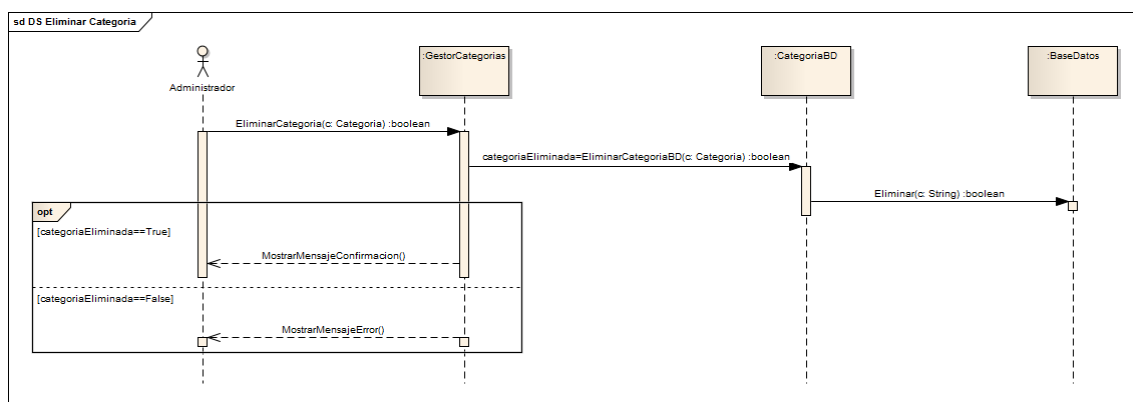


Diagrama de secuencia: InsertarCategorías

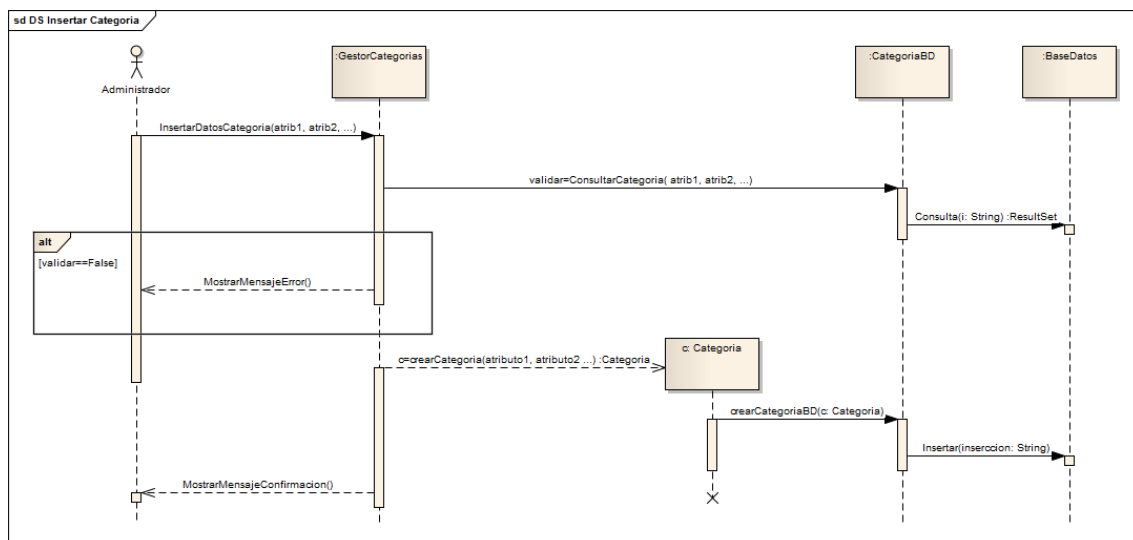


Diagrama de secuencia: ModificarCategorías

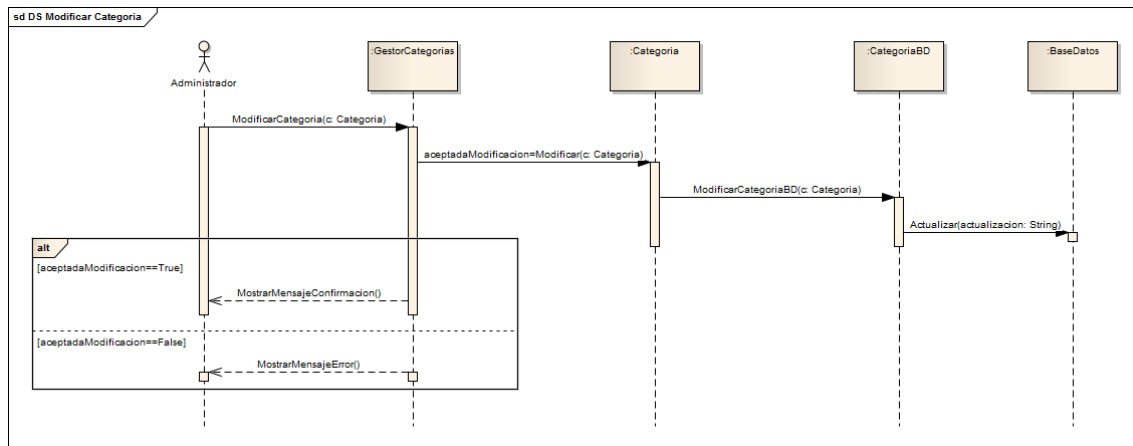


Diagrama de secuencia: ConsultarInstalaciones

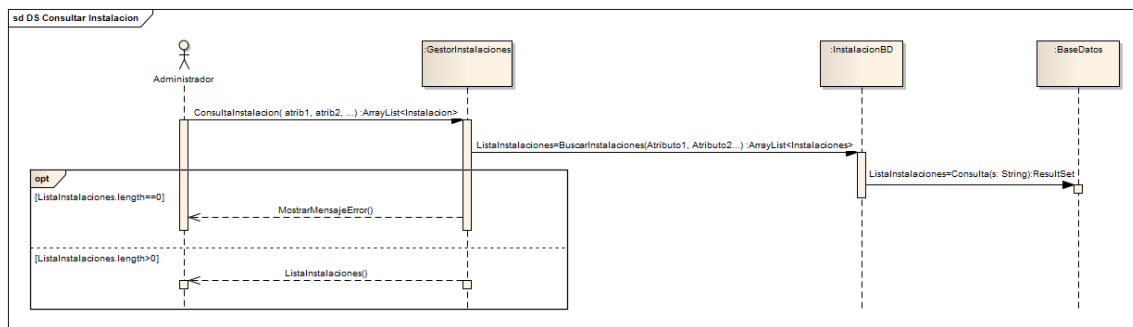


Diagrama de secuencia: EliminarInstalaciones

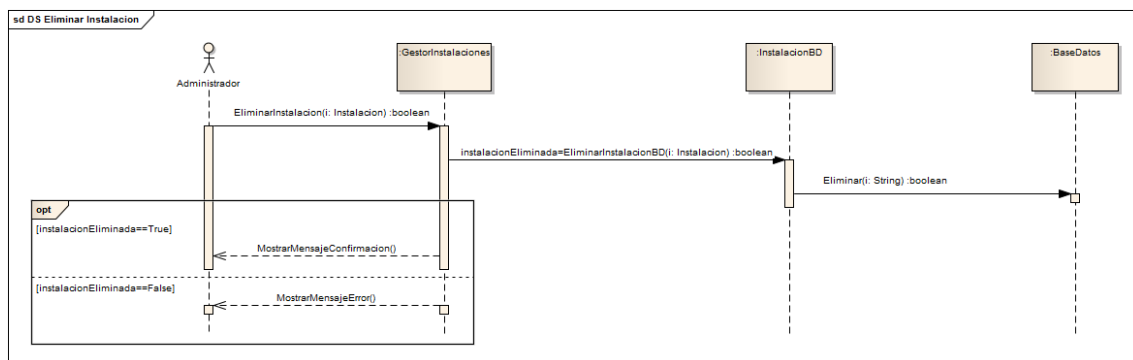


Diagrama de secuencia: InsertarInstalaciones

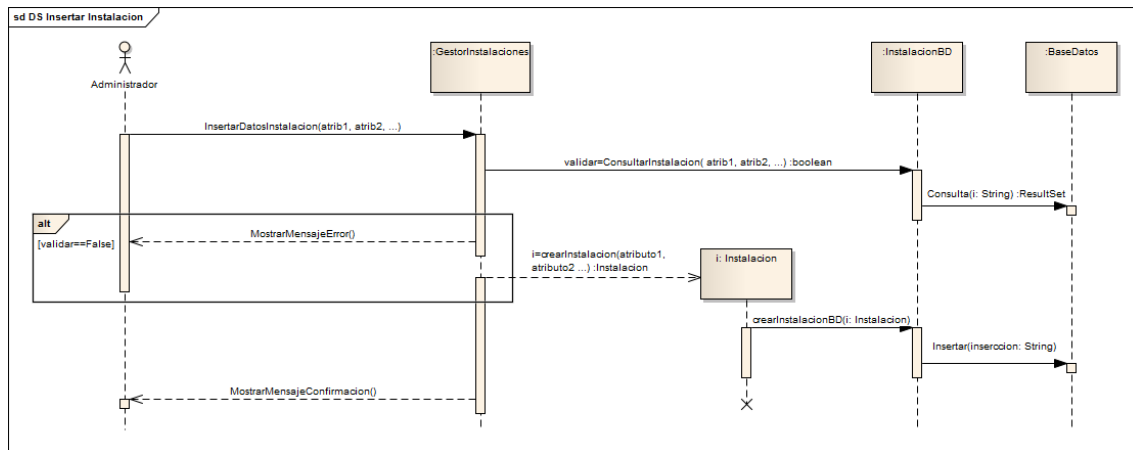


Diagrama de secuencia: ModificarInstalaciones

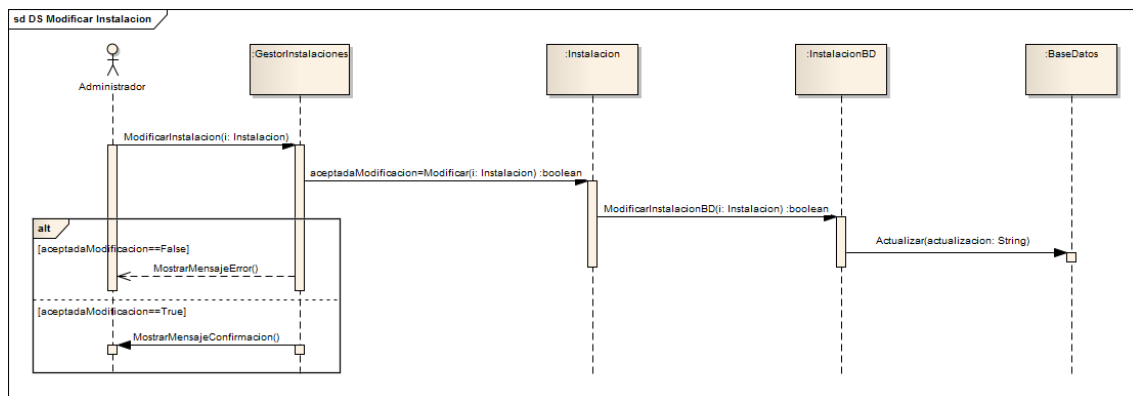


Diagrama de secuencia: ConsultarPago Actividad

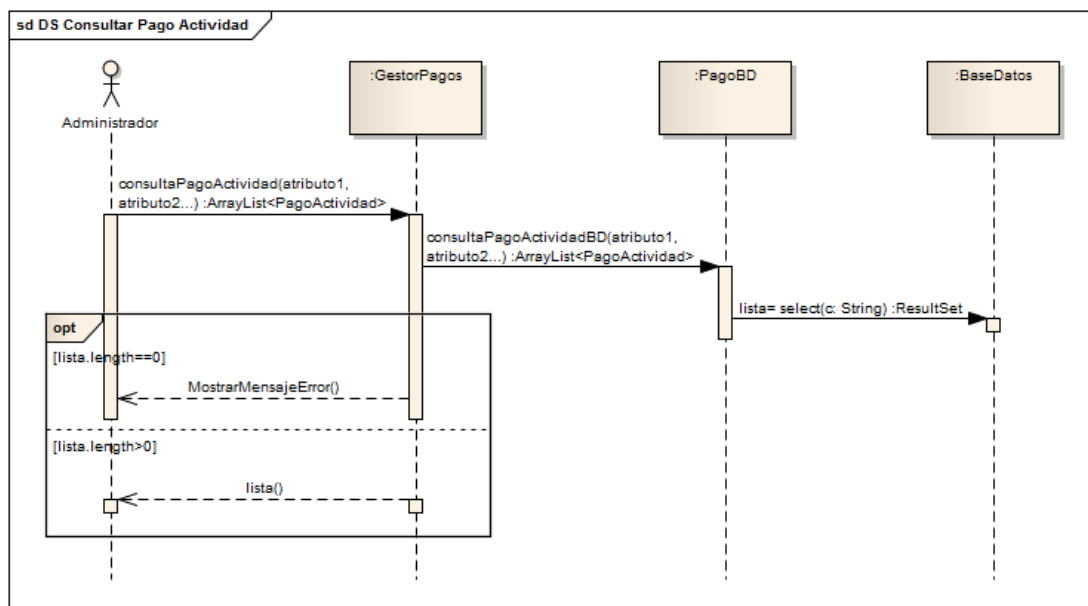


Diagrama de secuencia: EliminarPago Actividad

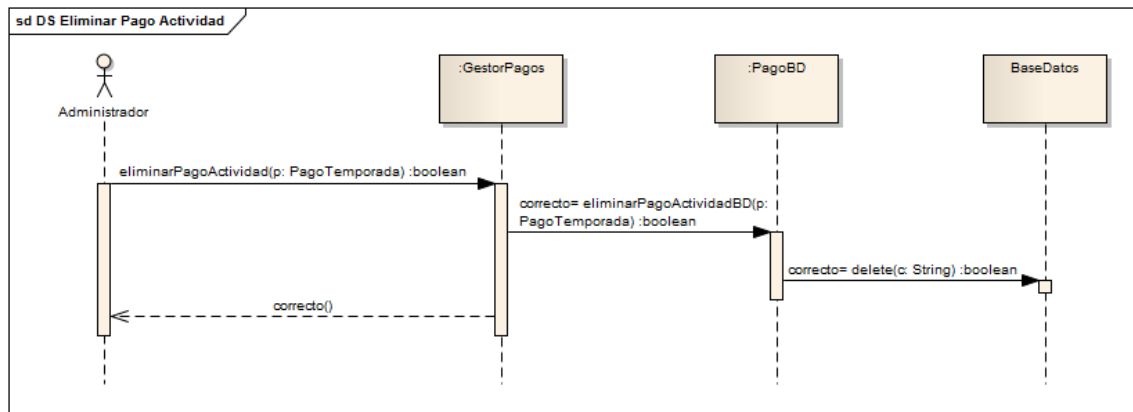


Diagrama de secuencia: InsertarPago Actividad

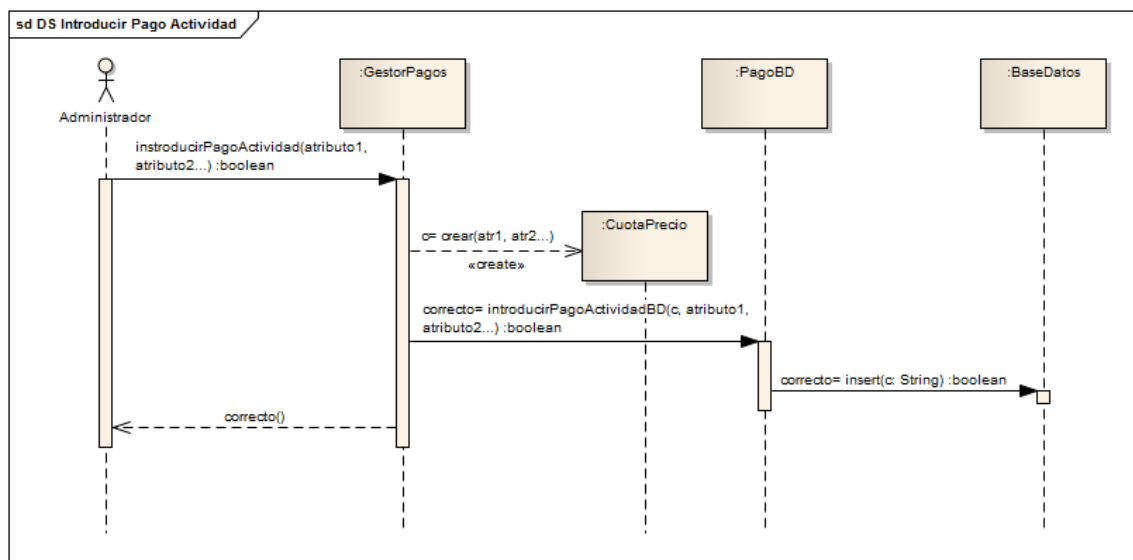


Diagrama de secuencia: ModificarPago Actividad

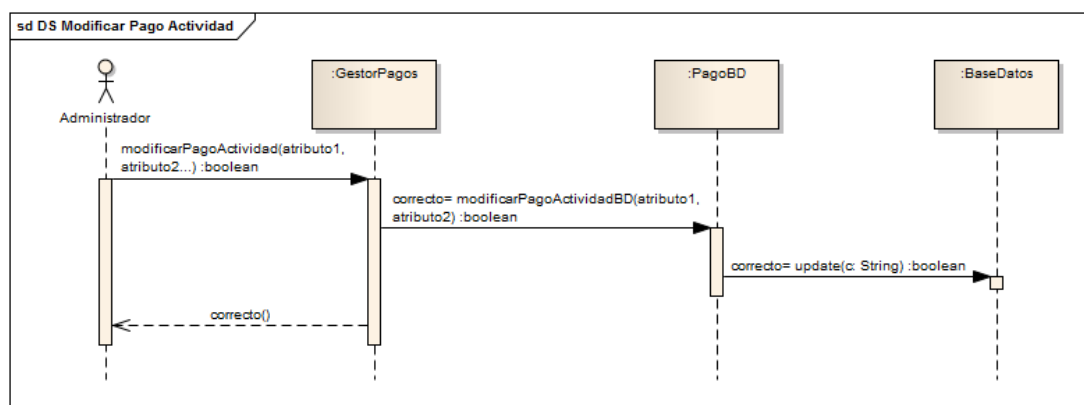


Diagrama de secuencia: ConsultarPago Temporada

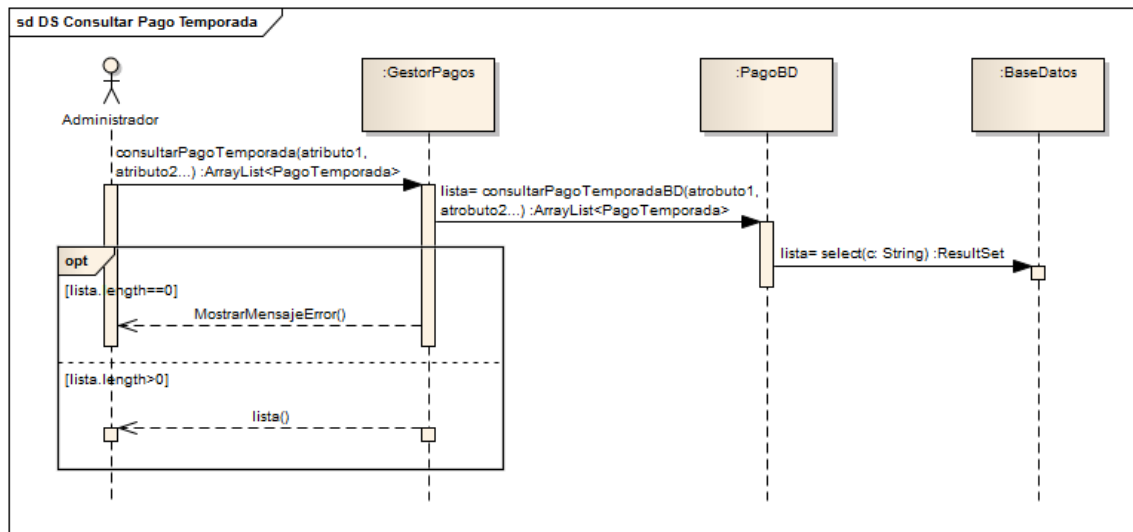


Diagrama de secuencia: EliminarPago Temporada

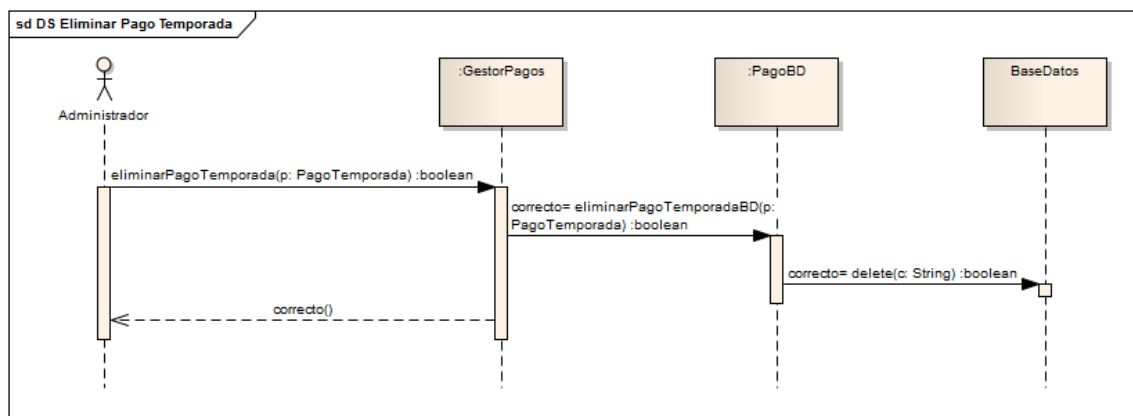


Diagrama de secuencia: InsertarPago Temporada

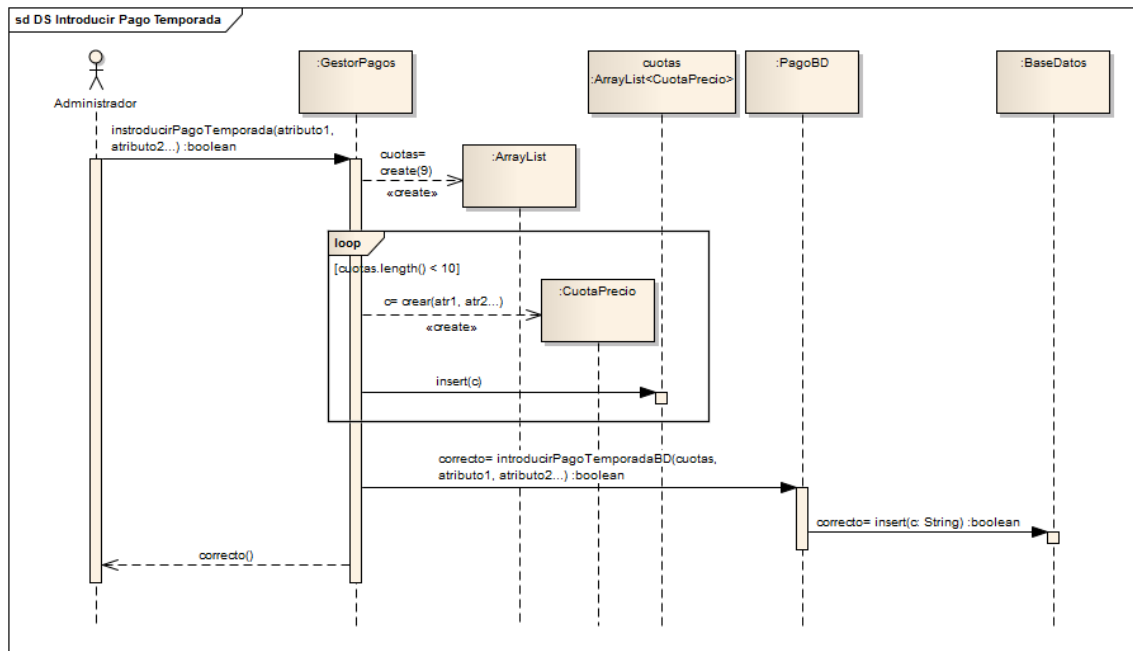


Diagrama de secuencia: ModificarPago Temporada

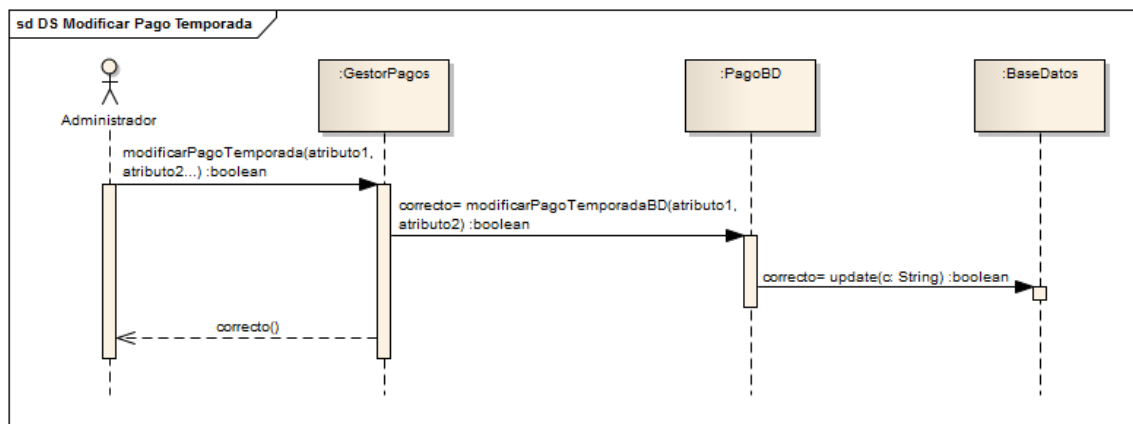


Diagrama de secuencia: ConsultarEquipos

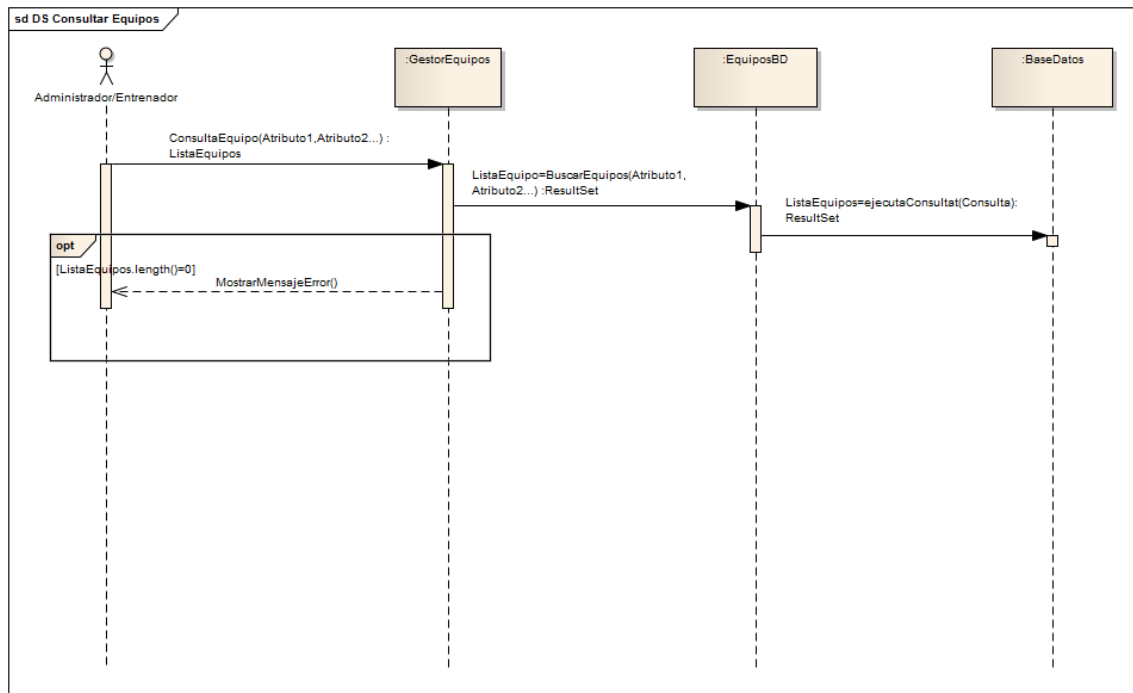


Diagrama de secuencia: EliminarEquipos

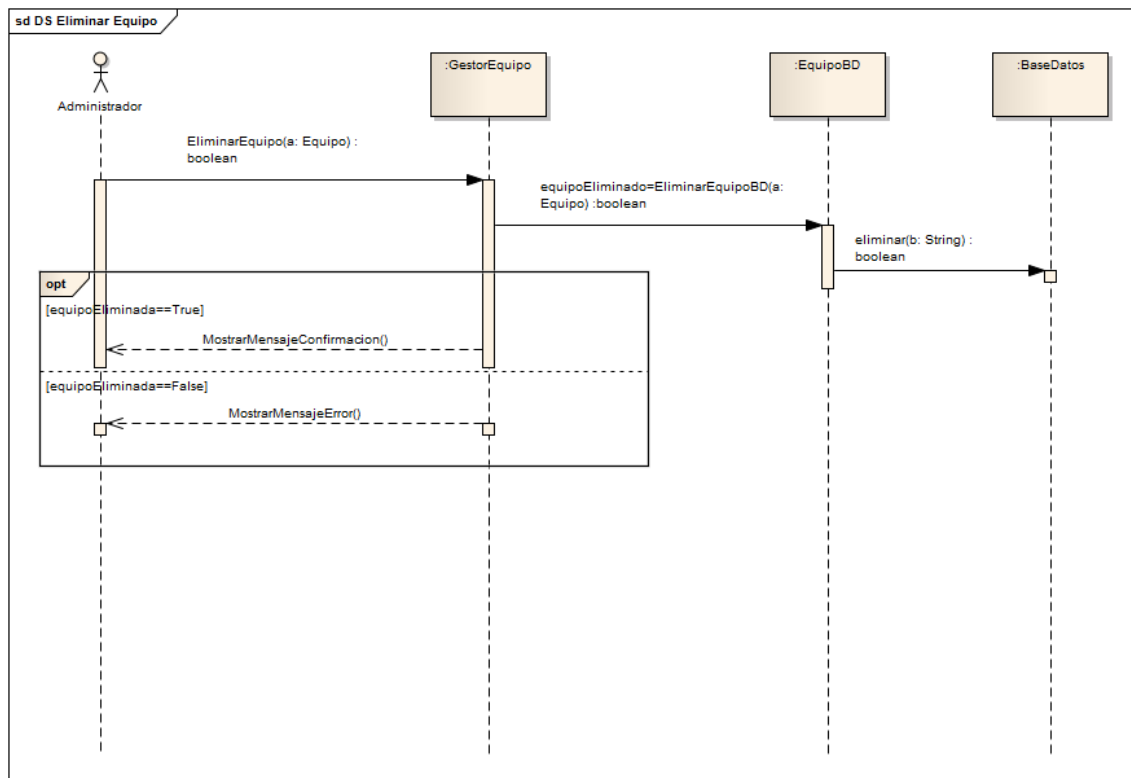


Diagrama de secuencia: InsertarEquipos

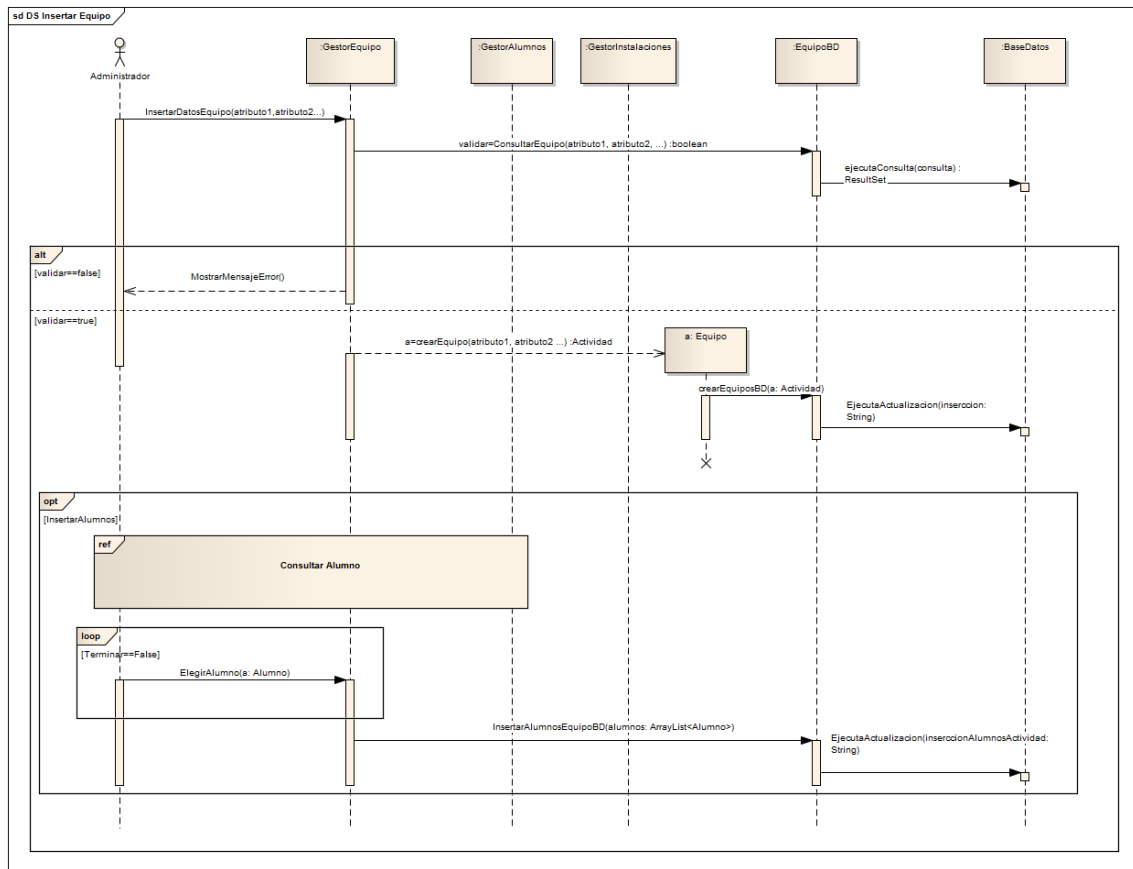


Diagrama de secuencia: ModificarEquipos

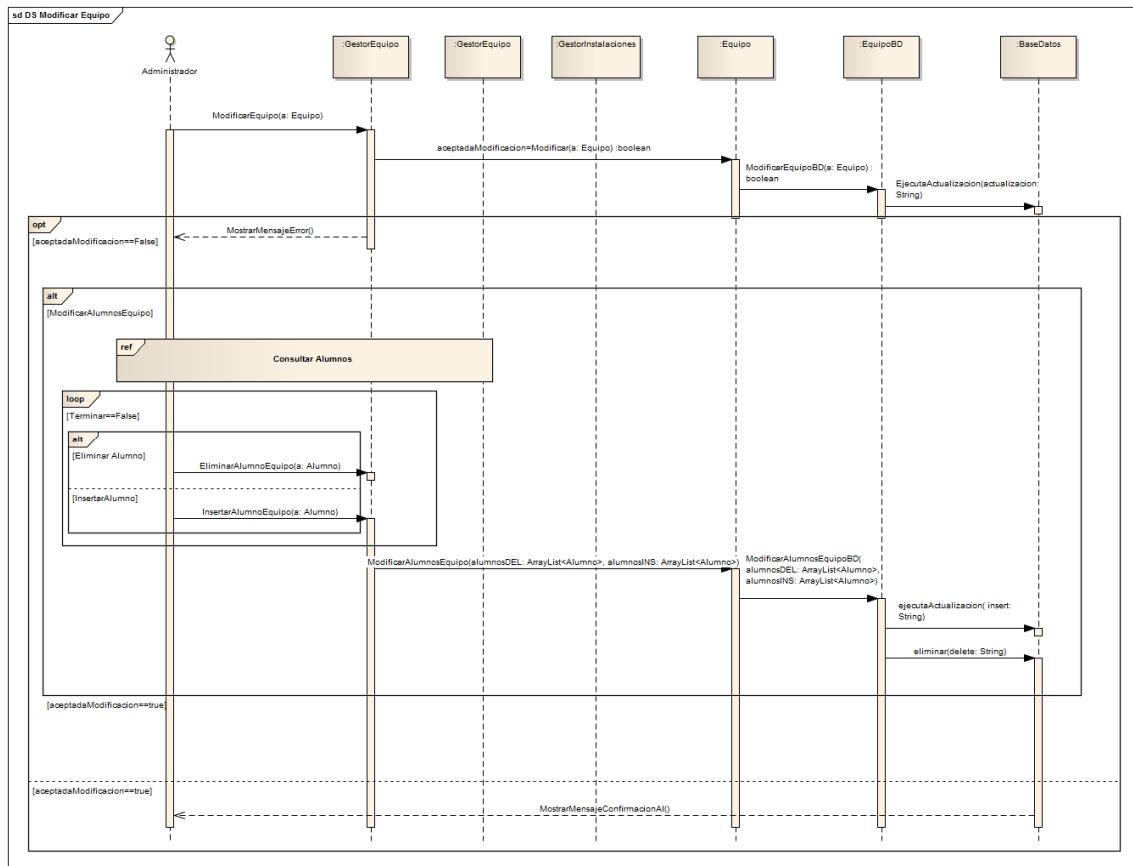


Diagrama de secuencia: ConsultarGrupos

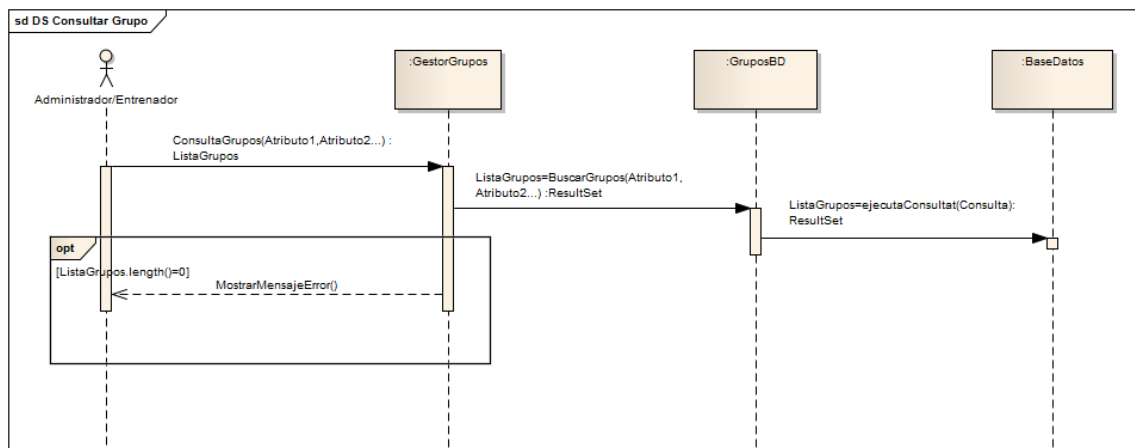


Diagrama de secuencia: EliminarGrupos

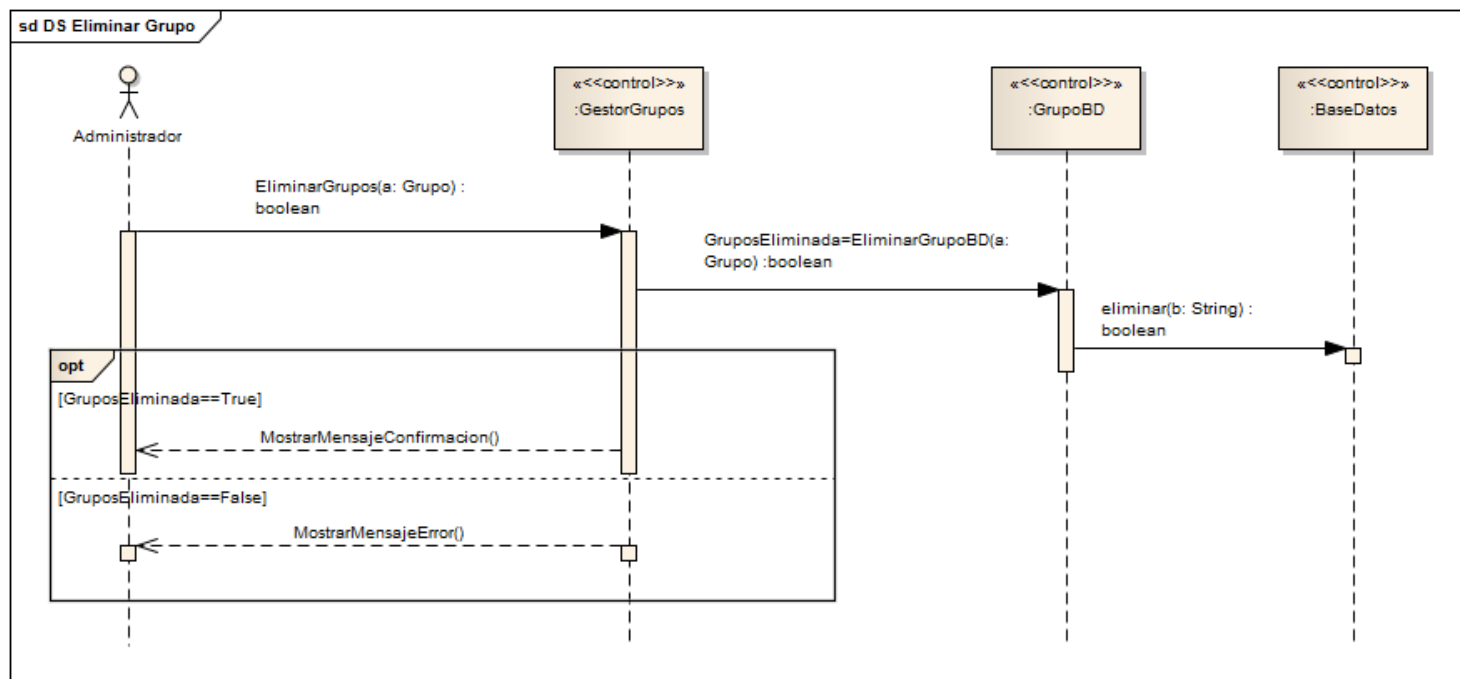


Diagrama de secuencia: InsertarGrupos

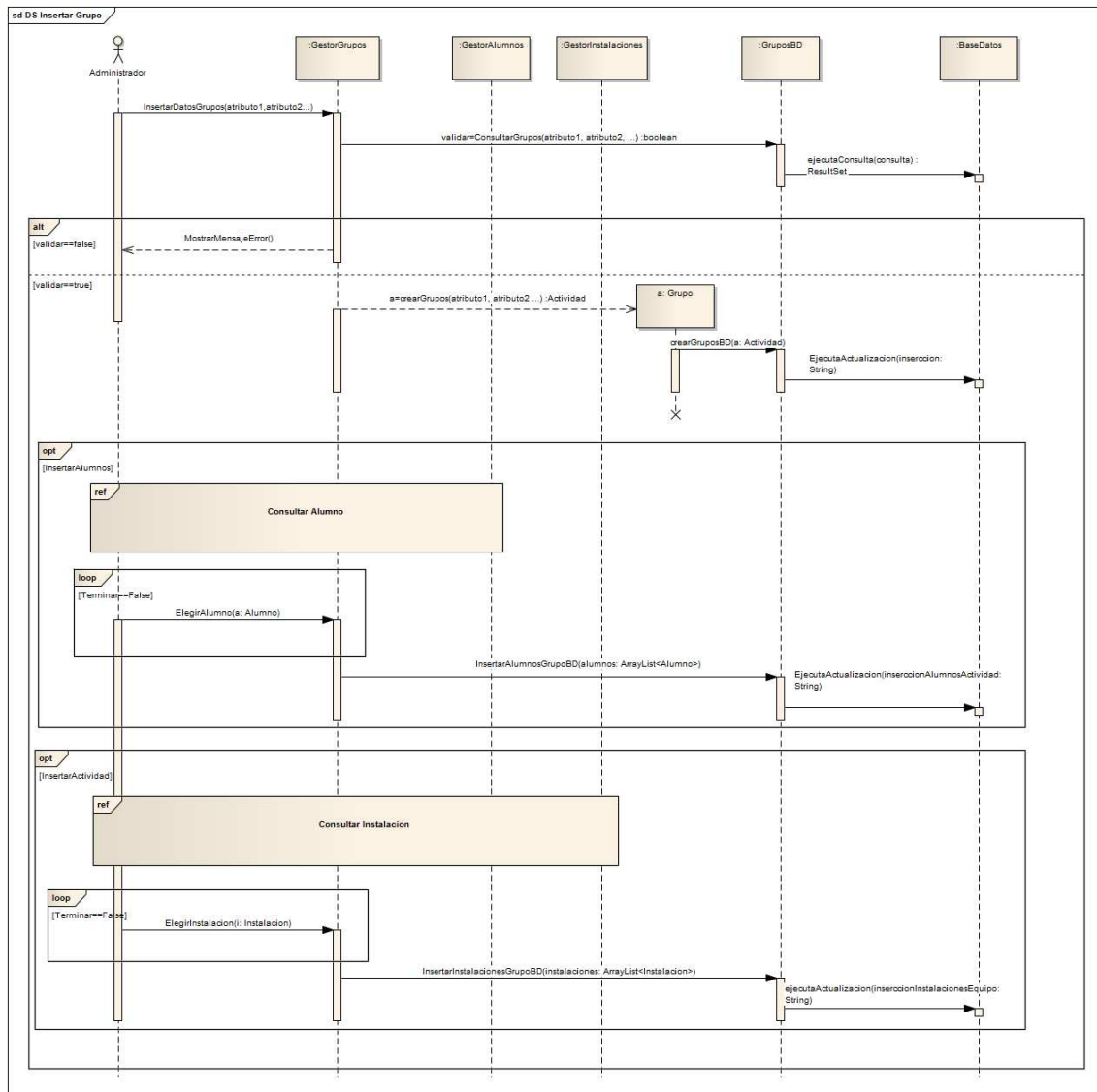


Diagrama de secuencia: ModificarGrupos

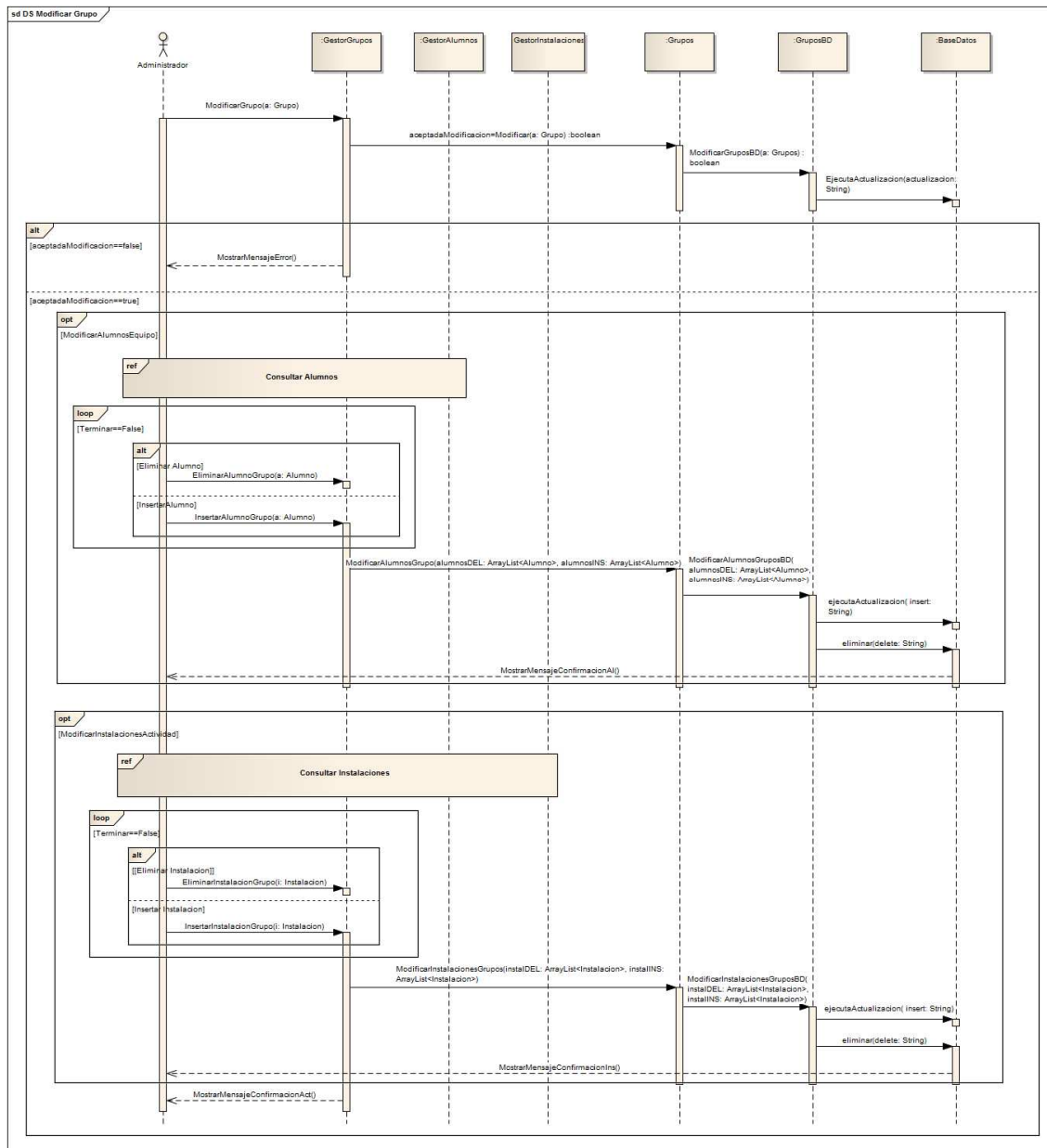


Diagrama de secuencia: ConsultarTemporadas

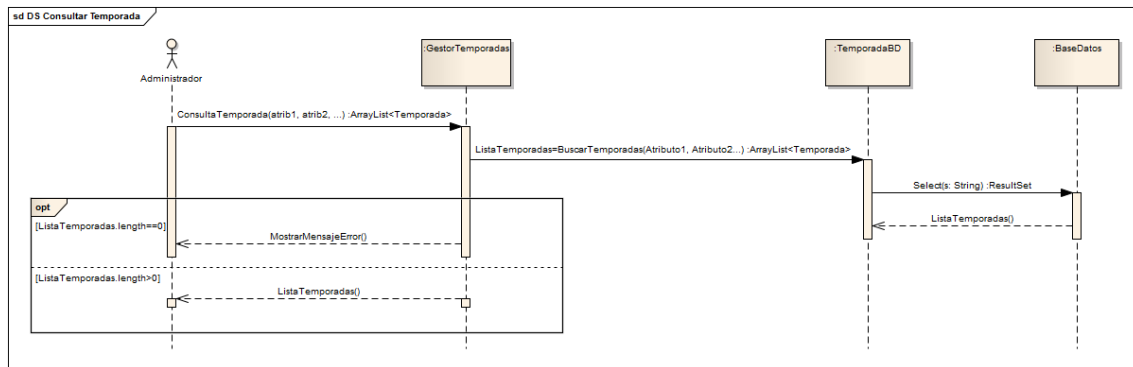


Diagrama de secuencia: EliminarTemporadas

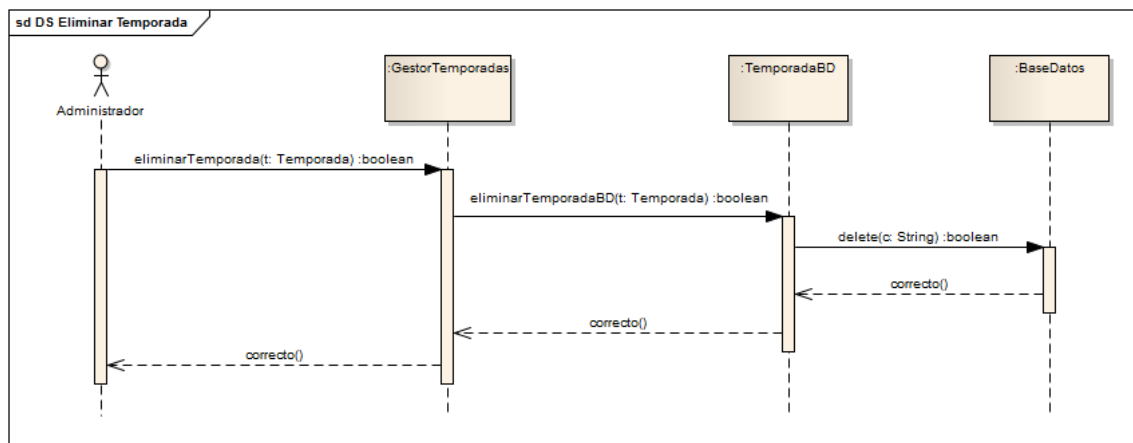


Diagrama de secuencia: InsertarTemporadas

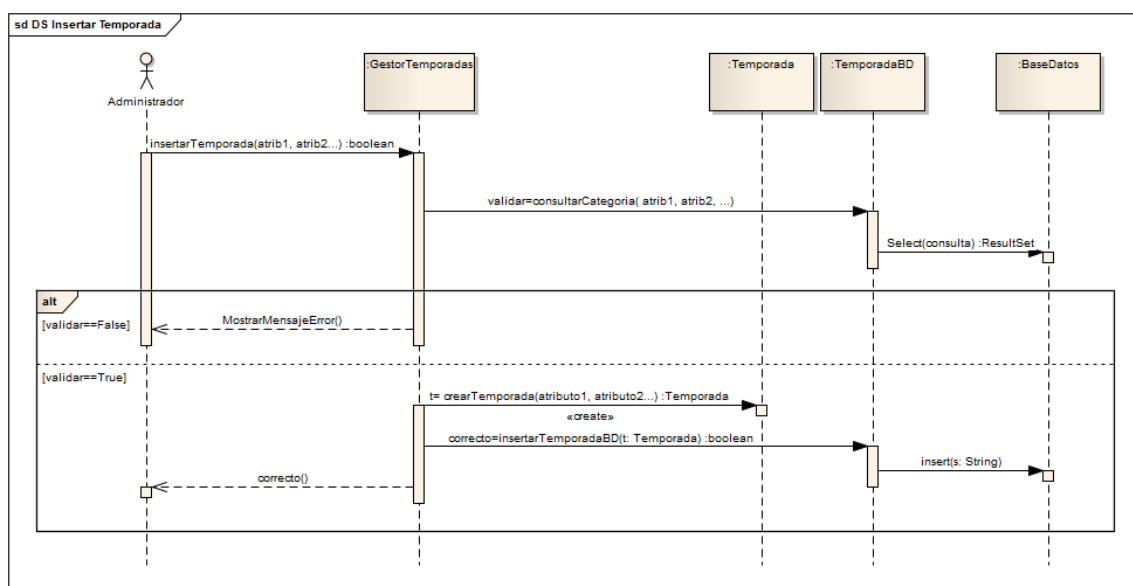
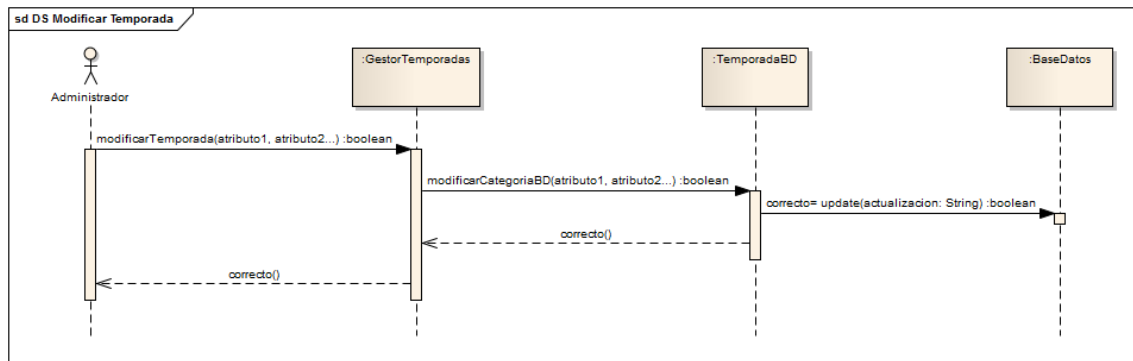
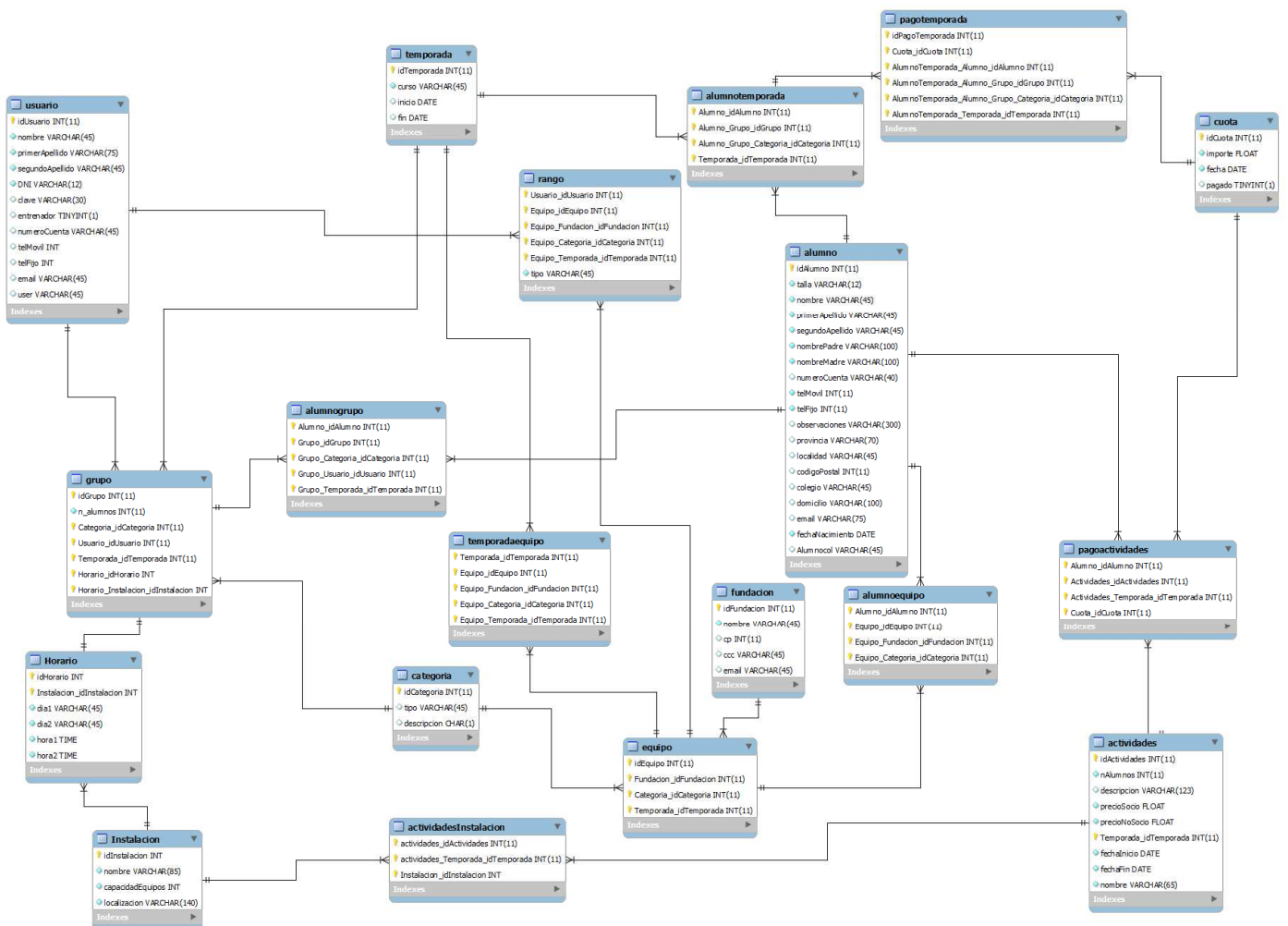


Diagrama de secuencia: ModificarTemporadas



Paso de clases a modelo relacional



Modelar Diagrama de Componentes

El diagrama de paquetes está estructurado en agrupaciones lógicas, pero es necesario definir subsistemas, es decir, cerrar las partes reutilizables del sistema en componentes. Para ello es necesario añadir interfaces.

Con el diagrama de componentes se va a definir la implementación del sistema a partir de los módulos de software y su interrelación.

Con cada componente se va a representar una parte modular de un sistema que encapsula sus contenidos y cuya manifestación es reemplazable.

Cada componente puede:

- Tener atributos y operaciones.
- Representar a cualquier cosa desde clases sencillas a aplicaciones, subsistemas y sistemas.
- Tener interfaces proporcionadas, requeridas y puertos, es decir, servicios que ofrece un componente a otro o que necesita de otro.

Las interfaces permiten a cada componente comunicarse con otros componentes. Estas son una colección de operaciones que se utilizan para especificar un servicio de un componente.

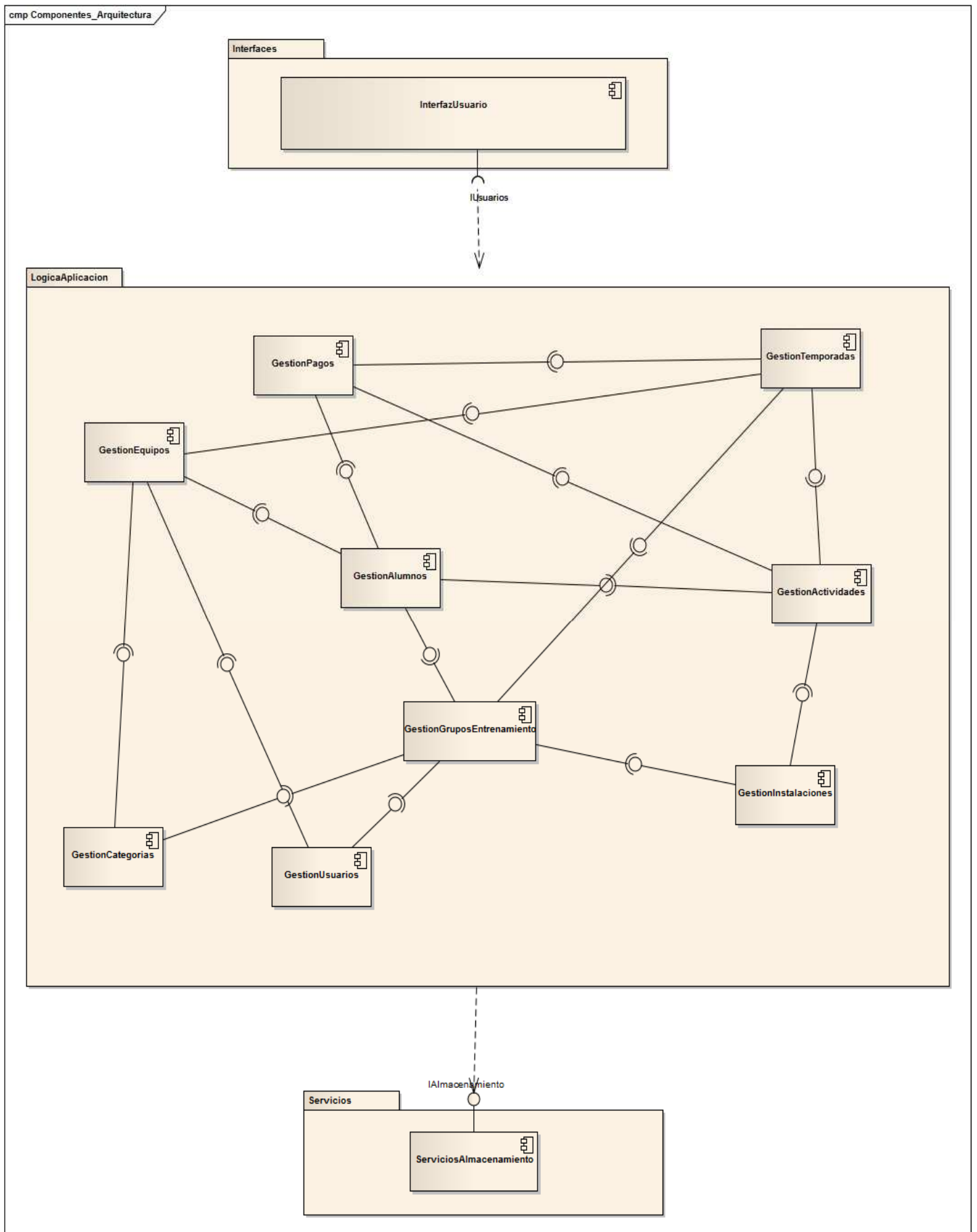
Las interfaces facilitan la sustitución y reutilización de componentes:

- Un componente puede exportar su interfaz e importar interfaces de más de un componente.
- Un componente que utiliza una interfaz determinada funcionará adecuadamente independientemente del componente que realice la interfaz.
- El componente que realiza la interfaz es siempre sustituible por un componente o conjunto de componentes que implementen dicha interfaz.
- Un componente puede utilizarse en un contexto determinado si y solo si todas sus interfaces de importación son suministradas por otros componentes.

Un puerto de componente agrupa un conjunto semánticamente cohesivo de interfaces proporcionadas y requeridas.

El diagrama de paquetes está estructurado en agrupaciones lógicas, pero necesitamos definir subsistemas, es decir, cerrar partes del sistema en componentes.

Diagrama de Componentes



Encajar el Diagrama de Clases (obtenido anteriormente) en la arquitectura obtenida en el apartado anterior

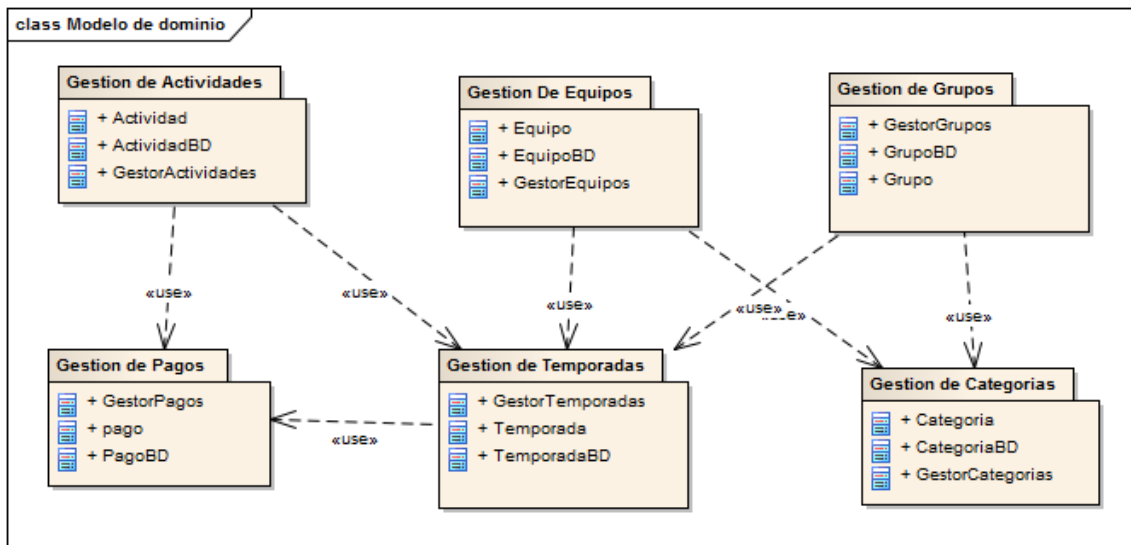


Diagrama de Despliegue de Diseño

Para obtener el diagrama de despliegue de diseño se determinan los posibles nodos del sistema, con los cuales se especifica el hardware físico sobre el que se ejecuta el sistema de software y cómo se ubica el software en el hardware. Además se especifican los componentes y las conexiones entre los distintos nodos.

Se tienen los siguientes nodos, los cuales pueden ser dispositivos físicos o entornos de ejecución:

- PC, que es el ordenador que hay en la oficina.
- PC (Portatil), que es un portátil que pueden usar los entrenadores durante los partidos.
- Aplicación Fundación, que es la aplicación sobre la que se trabaja.
- Servidor de Persistencia, que es la tecnología que se usará para almacenamiento de datos.

Dentro de los nodos anteriores se anidan los siguientes nodos con el prototipo <<ExecutionEnvironment>>, lo cual representa un tipo de entorno de ejecución para software.

- Execution Environment Windows 7
- Execution Environment MySQL

Diagrama de despliegue

