



ESCUELA POLITÉCNICA
SUPERIOR DE CÓRDOBA
Universidad de Córdoba



GRADO DE INGENIERÍA INFORMÁTICA

INGENIERÍA DEL SOFTWARE

DOCUMENTACIÓN DEL SISTEMA DE GESTIÓN DE ALUMNOS

Información sobre el sistema de gestión de alumnos

Autores:

*Luque León, Manuel Alejandro
Márquez Doblas, José
Rodríguez Guillén, Pablo*

FECHA: **23-12-2018**

ÍNDICE DE CONTENIDOS

1.	INTRODUCCIÓN.....	1
2.	EXTRACCIÓN DE REQUISITOS.....	2
3.	CASOS DE USO.....	4
3.1.	Insertar Alumno.....	4
3.2.	Modificar Alumno.....	4
3.3.	Borrar Alumno.....	5
3.4.	Mostrar Alumno.....	6
3.5.	Especificar Líder.....	6
3.6.	Generar Backup.....	7
3.7.	Formatear BD.....	8
3.8.	Buscar Alumnos.....	9
3.9.	Cargar Backup.....	10
3.10.	Crear Coordinador.....	10
3.11.	Modificar Coordinador.....	11
3.12.	Insertar Ayudante.....	12
3.13.	Modificar Ayudante.....	12
3.14.	Eliminar Ayudante.....	13
4.	HISTORIAS DE USUARIO.....	14
4.1.	Insertar Alumno.....	14
4.2.	Modificar Alumno.....	15
4.3.	Borrar Alumno.....	16
4.4.	Mostrar Alumno.....	16
4.5.	Especificar Líder.....	17
4.6.	Generar Backup.....	17
4.7.	Formatear BD.....	18
4.8.	Buscar Alumnos.....	18
4.9.	Cargar Backup.....	19
4.10.	Crear Coordinador.....	20
4.11.	Modificar Coordinador.....	20
4.12.	Insertar Ayudante.....	21
4.13.	Modificar Ayudante.....	22

4.14.	Eliminar Ayudante.....	22
5.	DIAGRAMA DE CLASES.....	23
6.	DIAGRAMAS DE SECUENCIA.....	25
6.1.	Insertar Alumno.....	28
6.2.	Modificar Alumno.....	29
6.3.	Borrar Alumno.....	30
6.4.	Mostrar Alumnos.....	30
6.5.	Especificar Líder.....	31
6.6.	Generar Backup.....	32
6.7.	Formatear BD.....	33
6.8.	Buscar Alumno.....	34
6.9.	Cargar Backup.....	35
6.10.	Crear Coordinador.....	36
6.11.	Modificar Profesor.....	37
6.12.	Insertar Ayudante.....	38
6.13.	Modificar Ayudante.....	39
6.14.	Eliminar Ayudante.....	40
7.	PRODUCT BACKLOG.....	41
8.	SPRINTS BACKLOG.....	42
8.1.	Primera entrega.....	42
8.2.	Segunda entrega.....	44
9.	BURDOWN CHART.....	45
10.	MATRICES DE TRAZABILIDAD.....	46

1

INTRODUCCIÓN

La presente documentación ha sido desarrollada por Manuel Alejandro Luque León, José Márquez Doblas y Pablo Rodríguez Guillén para la asignatura Ingeniería del Software de Ingeniería en Informática de la Universidad de Córdoba.

El objetivo de este documento es facilitar la comprensión del funcionamiento del sistema de gestión de alumnos desarrollado para las prácticas de dicha asignatura. El objetivo de la práctica es desarrollar un software capaz de gestionar un grupo de alumnos con distintos roles de profesores.

El sistema está desarrollado en C++ y para el sistema operativo Ubuntu.

2

EXTRACCIÓN DE REQUISITOS

Actores

- * *Principal*: Profesor.
- * *Secundarios*: Alumnos.

Datos

- * DNI.*
- * Nombre.*
- * Apellidos.*
- * Teléfono.
- * Email corporativo.
- * Dirección postal.
- * Último curso matriculado.
- * Fecha de nacimiento.
- * Nº de equipo.
- * Líder (*Sí/No*).

> *Nota:* Los atributos marcados con un * serán campos obligatorios.

Extracción Requisitos

Requisitos funcionales

1. Insertar alumno. (1)
2. Modificarlo. (4)
3. Borrarlo. (4)
4. Mostrarlo. (4)
5. Insertar líder. (2)
6. Generar copia de seguridad. (5)
7. Restaurar copia de datos (5)
8. Formateo de base de datos. (5)
9. Buscar alumnos. (3)
10. Crear coordinador. (1)
11. Modificar coordinador. (4)
12. Añadir profesor ayudante. (2)
13. Modificar profesor ayudante. (4)
14. Eliminar profesor ayudante. (4)

> **Nota:* Las prioridades se establecerán con un número entre paréntesis tras el requisito al que hace referencia (siendo 1 la máxima prioridad y 5 la mínima)

Requisitos no funcionales

1. SO: linux.
2. C++.
3. Copia de seguridad fichero binario.
4. Mostrar datos en html.
5. Campo obligatorio de un alumno: dni, nombre, apellido.
6. Máximo 150 alumnos.

3

CASOS DE USO

3.1. Insertar Alumno

ID: 001.

Breve descripción: Permite insertar un nuevo alumno.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El máximo permitido será de 150 alumnos.
2. El alumno no debe existir.

Flujo Principal:

1. El caso empieza cuando el usuario desea añadir un alumno.
2. El sistema recoge los datos introducidos por el usuario.

Postcondiciones:

* El sistema guarda los datos introducidos.

Flujo alternativo:

- * Si no se introducen los campos obligatorios (DNI, nombre y apellidos), el sistema muestra un error y pide que se introduzcan.
- * Si el alumno existe se mostrará un error y se vuelve al menú.

3.2. Modificar Alumno

ID: 002.

Breve descripción: Permite editar un alumno ya existente.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El alumno debe existir.
2. El usuario debe haber buscado y seleccionado los alumnos.

Flujo Principal:

1. El caso empieza cuando el usuario desea modificar un alumno en el menú que aparece tras el proceso de buscar y seleccionar alumno.
2. El sistema pregunta al usuario que parámetro desea editar.
3. El sistema recoge el nuevo valor introducido por el usuario.
4. El sistema pregunta si se desea editar algún otro parámetro.
5. En caso afirmativo, se vuelve al paso 2.

Postcondiciones:

- * El sistema sustituye los datos antiguos por los nuevos datos introducidos.

Flujo alternativo:

- * Si el alumno a modificar no existe, el sistema muestra un error.

3.3. Borrar Alumno

ID: 003.

Breve descripción: Permite borrar un alumno.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El alumno debe existir.

Flujo Principal:

1. El caso empieza cuando el usuario desea eliminar un alumno en el menú que aparece tras el proceso de buscar y seleccionar alumno.
2. El sistema pide una confirmación del borrado del alumno seleccionado.

Postcondiciones:

- * El alumno seleccionado es borrado del sistema.

Flujo alternativo:

- * Si el alumno a borrar no existe, el sistema muestra un error.
- * Si el usuario rechaza la confirmación de borrado se rechaza la petición de borrado.

3.4. Mostrar alumnos

ID: 004.

Breve descripción: Permite imprimir los datos en un fichero.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El alumno debe existir.
2. El usuario debe haber buscado y seleccionado los alumnos.
3. El usuario puede haber seleccionado la opción de ordenar la selección.

Flujo Principal:

1. El caso comienza cuando el usuario quiere mostrar los datos de uno o varios alumnos.
2. El usuario debe buscar y seleccionar los alumnos a mostrar.
3. El sistema obtiene y representa los datos de los alumnos siguiendo una plantilla predefinida.

Postcondiciones:

- * Los datos se imprimen en el fichero.

Flujo alternativo:

- * Si el alumno buscado no existe el sistema mostrará un mensaje de error.

3.5. Seleccionar líder

ID: 005.

Breve descripción: Permite introducir quien es el alumno líder de grupo (Solo 1 por grupo).

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El alumno debe existir.
2. No puede haber otro líder en ese grupo.

Flujo Principal:

1. El caso comienza cuando el usuario quiere seleccionar quien es el líder de un grupo.
2. El usuario debe seleccionar el grupo del que quiere seleccionar líder.
3. El usuario introduce al miembro del grupo a hacer líder.
4. El sistema comprueba que no hay otro líder en el grupo.

Postcondiciones:

- * El alumno queda guardado como líder de su grupo.

Flujo alternativo:

- * Si el alumno seleccionado no pertenece al grupo el sistema mostrará un mensaje de error.
- * Si ya hay un líder en ese grupo el sistema mostrará un error.

3.6. Generar Backup

ID: 006.

Breve descripción: Guardar un archivo binario con los datos de los alumnos.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. Debe existir al menos un alumno en la base de datos.
2. El usuario debe ser profesor coordinador.

Flujo Principal:

1. El caso comienza cuando el usuario quiere guardar una copia exacta de la base de datos.
2. El usuario seleccionará donde quiere guardar el archivo.

Postcondiciones:

- * El sistema crea el archivo.

Flujo alternativo:

- * Si no hay datos el sistema mostrará un mensaje de error.
- * Si no existe la ruta propuesta por el usuario el sistema mostrará un mensaje de error.
- * Si el usuario no es coordinador el sistema muestra un error y se vuelve al menú.

3.7. Formatear la Base de Datos

ID: 007.

Breve descripción: Borra todos los datos almacenados en la base de datos.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. La base de datos debe tener almacenada información de al menos un alumno.
2. El usuario debe ser profesor coordinador.

Flujo Principal:

1. El caso empieza cuando el usuario elige la opción de reset de la base de datos.
2. Se realiza una pregunta al usuario para asegurarse de que el usuario desea borrar todos los datos.

Postcondiciones:

- * El sistema borra todos los datos de la base de datos.

Flujo alternativo:

- * Si no se encuentra ningún alumno en la base de datos, se le informará al usuario y no se realizará ninguna acción.
- * Si el usuario no es coordinador el sistema muestra un error y se vuelve al menú.

3.8. Buscar alumnos

ID: 008.

Breve descripción: Busca uno, o un conjunto de alumnos y los selecciona.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El alumno o conjunto de alumnos a seleccionar debe existir.

Flujo Principal:

1. El caso empieza cuando el usuario selecciona la opción de buscar.
2. Se pregunta al usuario si desea seleccionar:
 - * Un alumno.
 - * Un equipo.
 - * Todos los alumnos.
 - * Un número determinado de alumnos.
3. Según lo que se haya seleccionado:
 - * *Un alumno:*
 - * Se le pregunta al usuario por el parámetro por el que se quiere buscar:
 - * DNI.
 - * Apellidos.
 - * Se pide dicho parámetro.
 - * *Un equipo:*
 - * Se pide el número del equipo.
 - * *Un numero determinado:*
 - * El usuario introduce dicho número.
 - * Se pregunta el parámetro por el que se quiere buscar (DNI/Apellidos).
 - * Se itera preguntando dicho parámetro el número de veces introducido previamente.

4. Se sale de la búsqueda

Postcondiciones:

- * El alumno o conjunto de alumnos de la búsqueda son seleccionados y se abre el menú para seleccionar la acción a realizar.

Flujo alternativo:

* Si en alguna búsqueda no se encuentra el alumno indicado, mostrar mensaje de error y dar la opción de seguir buscando, o de salir.

3.9. Cargar Backup

ID: 009.

Breve descripción: Lee un archivo binario con los datos de los alumnos.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El usuario debe ser profesor coordinador.

Flujo Principal:

1. El caso comienza cuando el usuario quiere cargar una copia de seguridad de la base de datos.
2. El usuario introduce donde está almacenado el archivo.

Postcondiciones:

- * El sistema carga la información del archivo.

Flujo alternativo:

- * Si la ruta introducida no corresponde a una copia de seguridad del sistema, este devolverá un mensaje de error.
- * Si el usuario no es coordinador el sistema muestra un error y se vuelve al menú.

3.10. Crear Coordinador

ID: 010.

Breve descripción: Permite insertar al profesor coordinador.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. Se ejecuta al iniciar el sistema por primera vez, por tanto el coordinador no debe existir.

Flujo Principal:

1. El caso empieza cuando el usuario inicia el sistema por primera vez.
2. El sistema comprueba que no existe ningún coordinador y pide que el usuario se registre como tal.

Postcondiciones:

* El sistema recoge los datos introducidos por el usuario y lo registra como coordinador.

Flujo alternativo:

* Si no se introducen los campos obligatorios (DNI, nombre, apellidos y contraseña), el sistema muestra un error y pide que se introduzcan.

3.11. Modificar Coordinador

ID: 011.

Breve descripción: Permite editar al coordinador de un sistema.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El coordinador debe existir.
2. El usuario debe identificarse con la clave del coordinador actual del sistema, en caso de ser un nuevo coordinar dicha clave debe haber sido provista por el centro.

Flujo Principal:

1. El caso empieza cuando el usuario desea modificar la información del coordinador, ya sea por corregir o añadir información del coordinador actual o por un cambio de coordinador.
2. El sistema pide la clave de coordinador para confirmar la selección de la opción.
3. El sistema pregunta al usuario que parámetro desea editar.
4. El sistema recoge el nuevo valor introducido por el ususario.
5. El sistema pregunta si se desea editar algún otro parámetro.
6. En caso afirmativo, se vuelve al paso 3.

Postcondiciones:

* El sistema sustituye los datos antiguos por los nuevos datos introducidos.

Flujo alternativo:

* Si el usuario introduce una clave errónea al confirmar la selección de esta opción el sistema mostrará un error y volvería al menú.

3.12. Añadir Ayudante

ID: 012.

Breve descripción: Permite insertar un nuevo profesor ayudante.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El usuario debe ser profesor coordinador.

Flujo Principal:

1. El caso empieza cuando el usuario selecciona dicha opción en el menú.
2. El sistema pide la información del nuevo profesor.
3. El usuario introduce los datos solicitados.

Postcondiciones:

* El sistema recoge los datos introducidos por el usuario y lo registra como coordinador.

Flujo alternativo:

* Si no se introducen los campos obligatorios (DNI, nombre, apellidos y contraseña), el sistema muestra un error y pide que se introduzcan.

3.13. Modificar Ayudante

ID: 013.

Breve descripción: Permite editar un profesor ayudante ya existente.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El ayudante debe existir.
2. El usuario debe ser profesor coordinador.

Flujo Principal:

1. El caso empieza cuando el usuario desea modificar un ayudante.
2. El sistema pide al usuario que identifique al ayudante a modificar mediante su DNI.
3. El sistema pregunta al usuario que parámetro desea editar.
4. El sistema recoge el nuevo valor introducido por el usuario.
5. El sistema pregunta si se desea editar algún otro parámetro.
6. En caso afirmativo, se vuelve al paso 3.

Postcondiciones:

- * El sistema sustituye la información antigua por la nueva introducida.

Flujo alternativo:

- * Si el ayudante a modificar no existe, el sistema muestra un error.

3.14. Eliminar Ayudante

ID: 014.

Breve descripción: Permite eliminar un profesor ayudante.

Actores principales: Profesor.

Actores secundarios: Alumnos.

Precondiciones:

1. El ayudante debe existir.
2. El usuario debe ser profesor coordinador.

Flujo Principal:

1. El caso empieza cuando el usuario desea eliminar un ayudante.
2. El sistema pide una confirmación del borrado del alumno seleccionado.

Postcondiciones:

* El ayudante seleccionado es eliminado del sistema.

Flujo alternativo:

* Si el ayudante a borrar no existe, el sistema muestra un error.

* Si el usuario rechaza la confirmación de borrado se rechaza la petición de borrado y se vuelve al menú.

4

HISTORIAS DE USUARIO

4.1. Insertar Alumno

(Anverso)

ID: 001 Insertar Alumno

Quiero poder añadir al sistema un nuevo alumno

Prioridad: 1

(Reverso)

* Quiero poder introducir todos los datos de un nuevo alumno:

- * DNI
- * Nombre
- * Apellidos
- * Teléfono
- * Email UCO
- * Dirección postal
- * Curso más alto en el que está matriculado
- * Fecha de nacimiento
- * Nº equipo
- * Líder (Sí/No)

* Introducir el equipo y especificar si es líder o no, no es obligatorio al añadir un alumno.

4.2. Modificar Alumno

(Anverso)

ID: 002 Modificar Alumno

Como usuario quiero poder modificar un alumno que ya está en la base de datos.

Prioridad: 4

(Reverso)

* Quiero poder modificar todos los datos de un alumno.

4.3. Borrar Alumno

(Anverso)

ID: 003 Borrar Alumno

Quiero poder borrar un alumno registrado en la base de datos.

Prioridad: 4

(Reverso)

* Realizar borrado de un alumno seleccionado y todos sus datos.

4.4. Mostrar Alumno

(Anverso)

ID: 004 Mostrar Alumno

Como usuario quiero poder mostrar alumnos

Prioridad: 4

(Reverso)

* Quiero poder mostrar:

- * Un único alumno.
- * Los alumnos de un equipo.
- * Todos los alumnos registrados.

* Quiero poder mostrar todos los datos del alumno mostrado.

* Quiero que se me impriman los datos en un único archivo pdf.

4.5. Insertar Líder

(Anverso)

ID: 005 Insertar Líder

Quiero poder seleccionar un miembro del grupo como líder

Prioridad: 2

(Reverso)

* Permitir la selección de un grupo, y posteriormente la asignación de un líder.

* Imposibilitar la selección de dos líderes para un mismo grupo.

4.6. Generar Copia de Seguridad

(Anverso)

ID: 006 Generar Copia de Seguridad

Como usuario quiero poder hacer copias de seguridad de mis datos.

Prioridad: 5

(Reverso)

* Quiero poder exportar todos los datos a un archivo.

* Se deben poder exportar los datos por grupos también.

4.7. Formatear Base de Datos

(Anverso)

ID: 007 Formatear Base de Datos

Quiero poder eliminar todos los alumnos registrados en el sistema

Prioridad: 5

(Reverso)

* Formatear sistema eliminando todos los alumnos registrados con sus datos.

4.8. Buscar Alumnos

(Anverso)

ID: 008 Buscar Alumnos

Quiero poder buscar un alumno registrado en la base de datos.

Prioridad: 3

(Reverso)

* Quiero poder realizar la búsqueda por:

* DNI.

* Apellidos (si hay 2 alumnos con los mismos apellidos, mostrar nombre también).

* Quiero poder realizar distintas opciones con el alumno buscado.

4.9. Cargar Backup

(Anverso)

ID: 009 Cargar Backup

Como usuario quiero poder cargar una copia de seguridad.

Prioridad: 5

(Reverso)

* Permitir restaurar los datos desde una copia de seguridad antes generada con Generar Backup.

4.10. Crear Coordinador

(Anverso)

ID: 010 Crear Coordinador

Quiero tener un profesor como coordinador del grupo.

Prioridad: 1

(Reverso)

* Debe haber unicamente un profesor coordinador, necesario para el funcionamiento del sistema.

* Quiero poder introducir todos los datos de un profesor:

- * DNI
- * Nombre
- * Apellidos
- * Contraseña
- * Teléfono
- * Email UCO
- * Dirección postal
- * Coordinador (Sí/No) = Sí

* Teléfono, email y dirección no son campos obligatorios al añadir un profesor.

4.11. Modificar Coordinador

(Anverso)

ID: 011 Modificar Coordinador

Como usuario quiero poder modificar al profesor coordinador que ya está en la base de datos.

Prioridad: 4

(Reverso)

* Quiero poder modificar todos los datos del profesor coordinador.

4.12. Insertar Ayudante

(Anverso)

ID: 012 Insertar Ayudante

Quiero poder añadir al sistema un nuevo profesor ayudante

Prioridad: 2

(Reverso)

* Quiero poder introducir todos los datos de un nuevo profesor:

* DNI

* Nombre

* Apellidos

* Contraseña

* Teléfono

* Email UCO

* Dirección postal

* Coordinador (Sí/No) = No

* Teléfono, email y dirección no son campos obligatorios al añadir un profesor.

4.13. Modificar Ayudante

(Anverso)

ID: 013 Modificar Ayudante

Como usuario quiero poder modificar un profesor ayudante que ya está en la base de datos.

Prioridad: 4

(Reverso)

* Quiero poder modificar todos los datos de un profesor ayudante.

4.14. Eliminar Ayudante

(Anverso)

ID: 014 Eliminar Ayudante

Quiero poder eliminar un profesor ayudante registrado en la base de datos.

Prioridad: 4

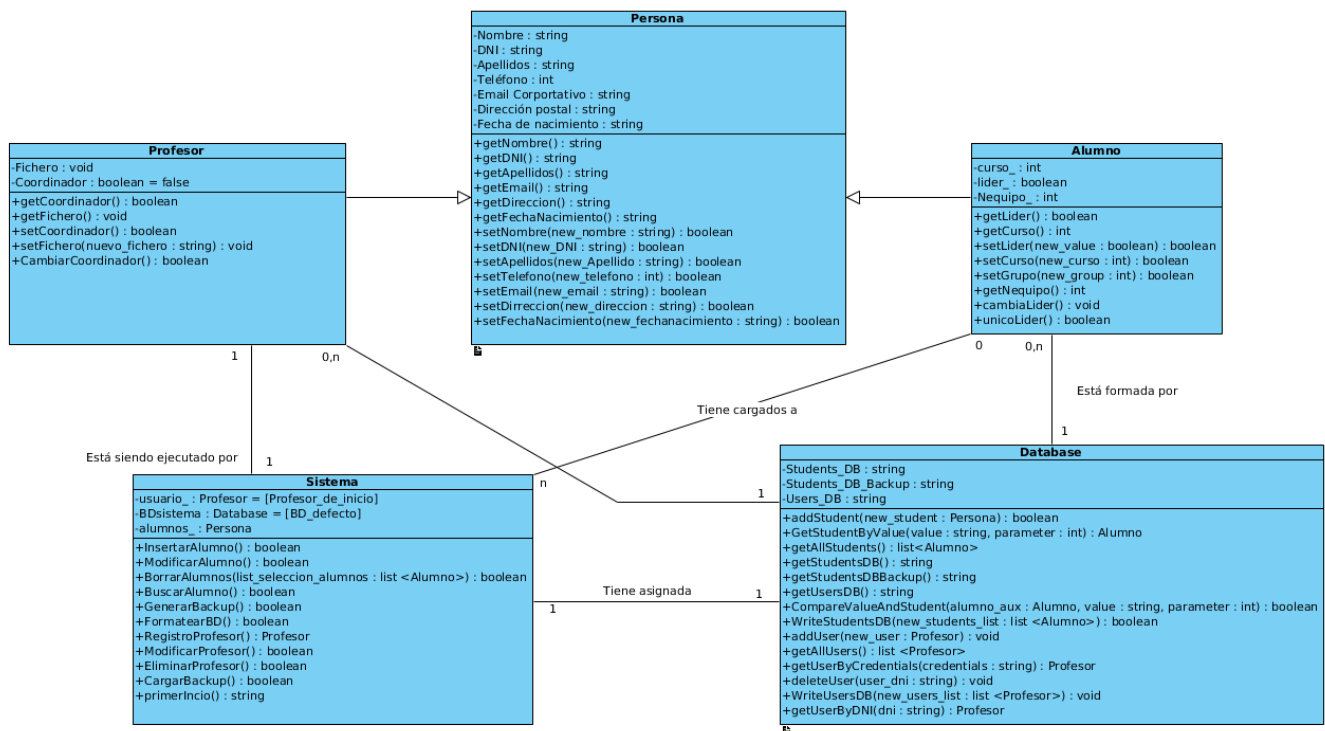
(Reverso)

* Eliminar un profesor ayudante seleccionado y todos sus datos.

5

DIAGRAMA DE CLASES

El sistema estará formado por distintas clases que se relacionarán entre sí para cumplir con todas las funcionalidades y requisitos del mismo.



La clase Persona, superclase de Alumno y Profesor, contiene los parámetros y métodos comunes a estos dos actores del sistema. Estos parámetros representan todos los datos que han de guardarse en la base de datos.

A estos atributos comunes agregamos, por parte de la clase Profesor, el nombre de su fichero de credenciales y un el atributo que señalará si ese profesor es coordinador o no. De igual forma, por parte de la clase Alumno, añadimos los atributos Grupo, UltimoCursoMatriculado y Lider. Este último señalará si el alumno es lider de su grupo o no.

La clase BD-sistema representa la base de datos que guarda los datos de los usuarios y los alumnos. Esta contiene los métodos necesarios para la gestión de los usuarios y los alumnos.

Por último la clase Sistema representa nuestro programa como tal y contiene los métodos que harán posible cumplir con los requisitos del mismo. Los atributos pertenecientes a esta clase guardan la relación con el profesor

conectado al sistema, las bases de datos activas y los alumnos con los que se está trabajando en ese momento.

6

DIAGRAMAS DE SECUENCIA

Este apartado describe una visión general de los diagramas de secuencia realizados que se encuentran en "Diagramas_de_Secuencia"

Elementos

Actor

Está presente en todos los diagramas de secuencia realizados.

Cuando se hace referencia al mismo con el identificador de "Profesor" se indica que el usuario que interactúa con el sistema puede ser tanto un coordinador como un ayudante.

Cuando se le identifica como coordinador se indica que solo el coordinador puede estar realizando la interacción con el sistema en cuestión.

Sistema

Está presente en todos los diagramas de secuencia realizados.

Se corresponde con la clase Sistema especificada en el diagrama de clases y es la única que interactúa con el usuario de forma directa.

En los diagramas, se señala que el usuario manda un mensaje de tipo llamada al sistema que se corresponde con una función, aunque esto no es directamente. El usuario elige una opción del menú que está directamente relacionada con un método de sistema.

Base de Datos

Está presente en algunos de los diagramas de secuencia. Esto se debe a que ciertos métodos de Sistema no necesitan interactuar con la base de datos.

Combined Fragments

Los representan los diferentes recuadros presentes en algunos diagramas de secuencia que incluyen dentro de ellos ciertos mensajes.

Se observan cuatro tipos:

Loop

Los mensajes de su interior iteran mientras que se cumpla la condición señalada entre corchetes.

Alt

Se utiliza para señalar una sentencia if. Implica que hay 2 secuencias de mensajes alternativas, que o se cumple una o la otra. La condición se señala entre corchetes cuando es necesario. En otras ocasiones se sobreentiende esta condición por el primer mensaje de la secuencia.

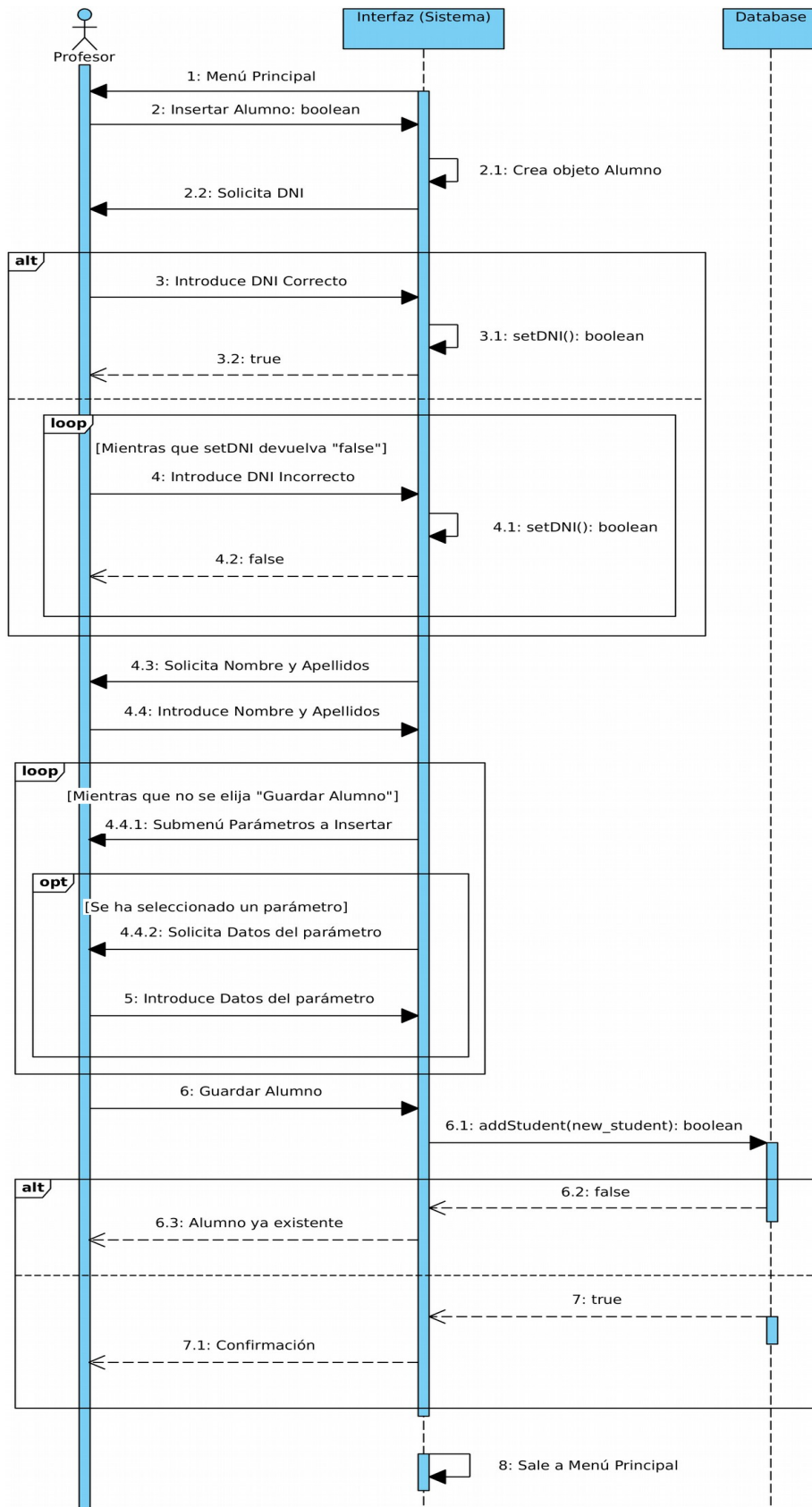
Opt

Implica que los mensajes que contiene no se producen en todas las ejecuciones, ya que son mensajes opcionales. En la mayoría de casos es utilizado para indicar el flujo de mensajes de una opción de un menú.

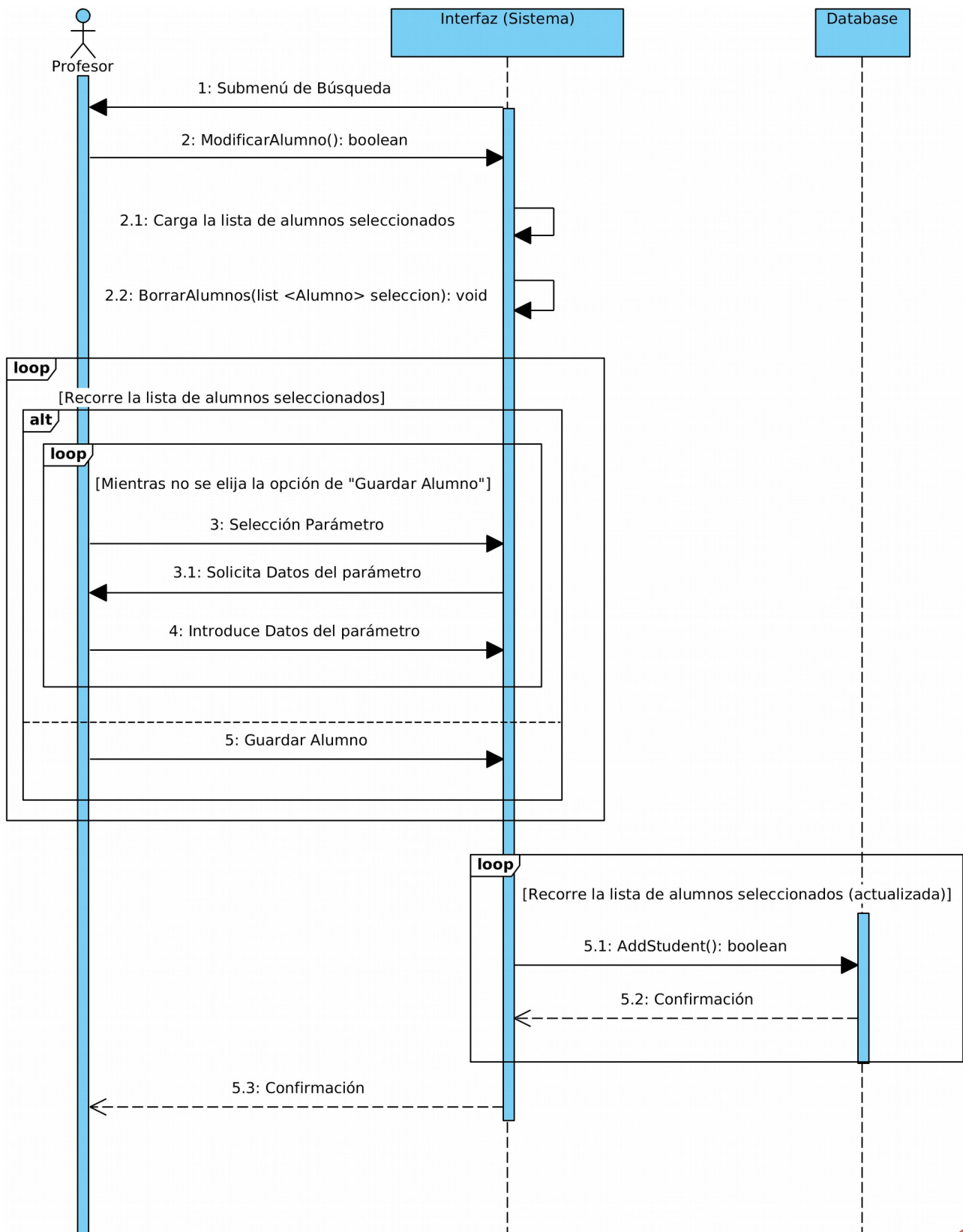
Break

Se incluye en el interior de algunos bucles para señalar que si se llega a esta secuencia, se produce una salida del mismo.

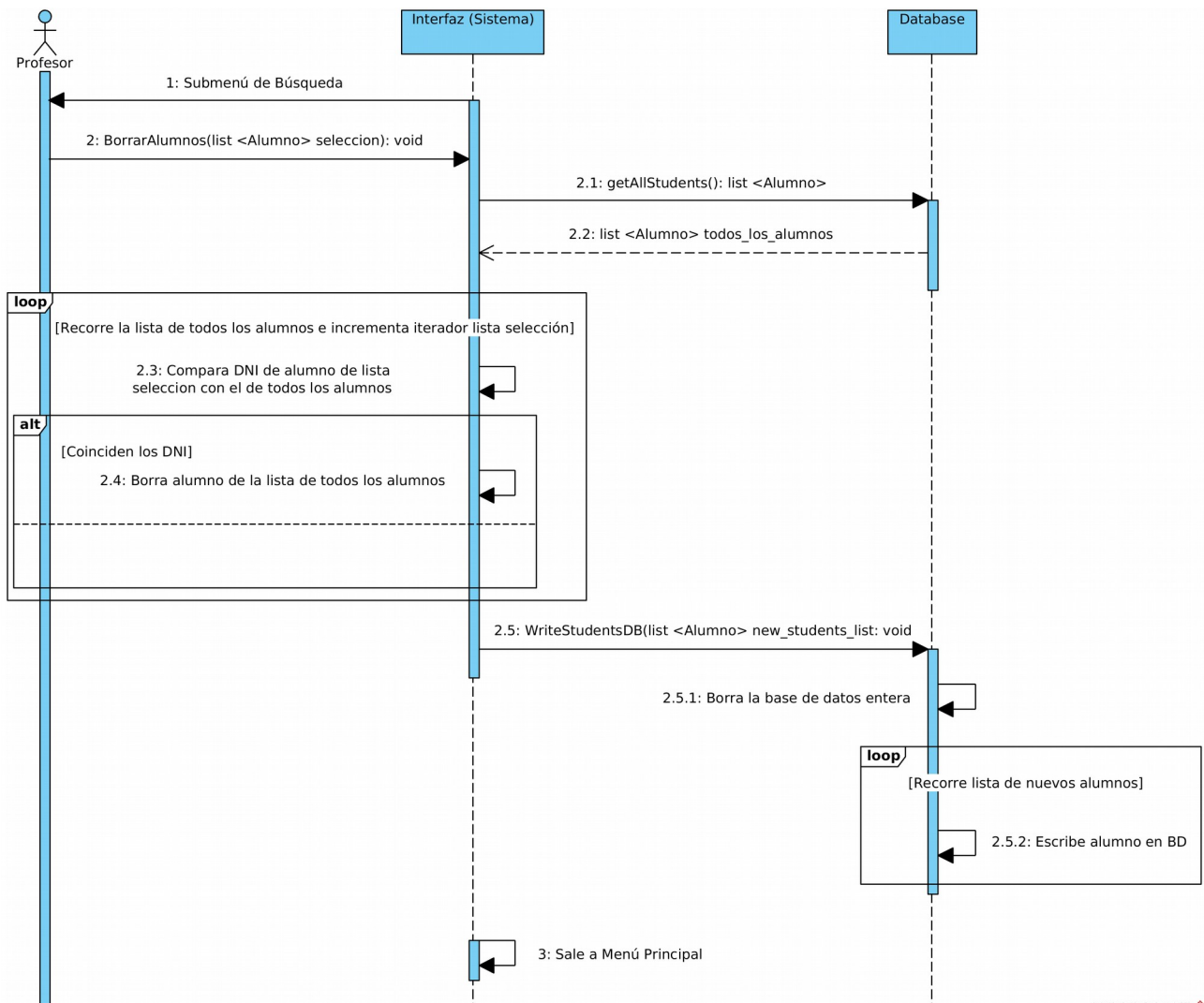
6.1. INSERTAR ALUMNO



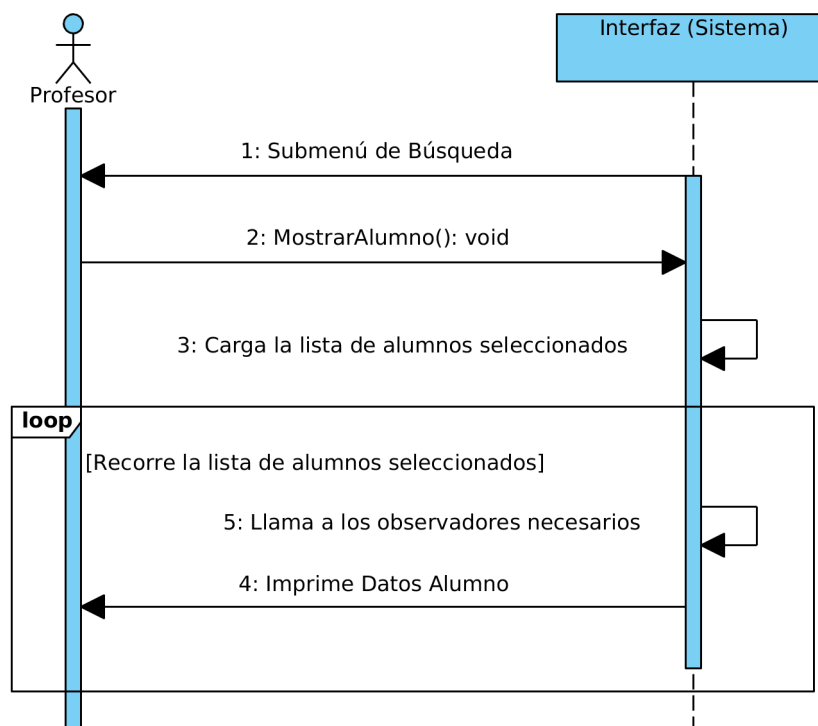
6.2. MODIFICAR ALUMNO



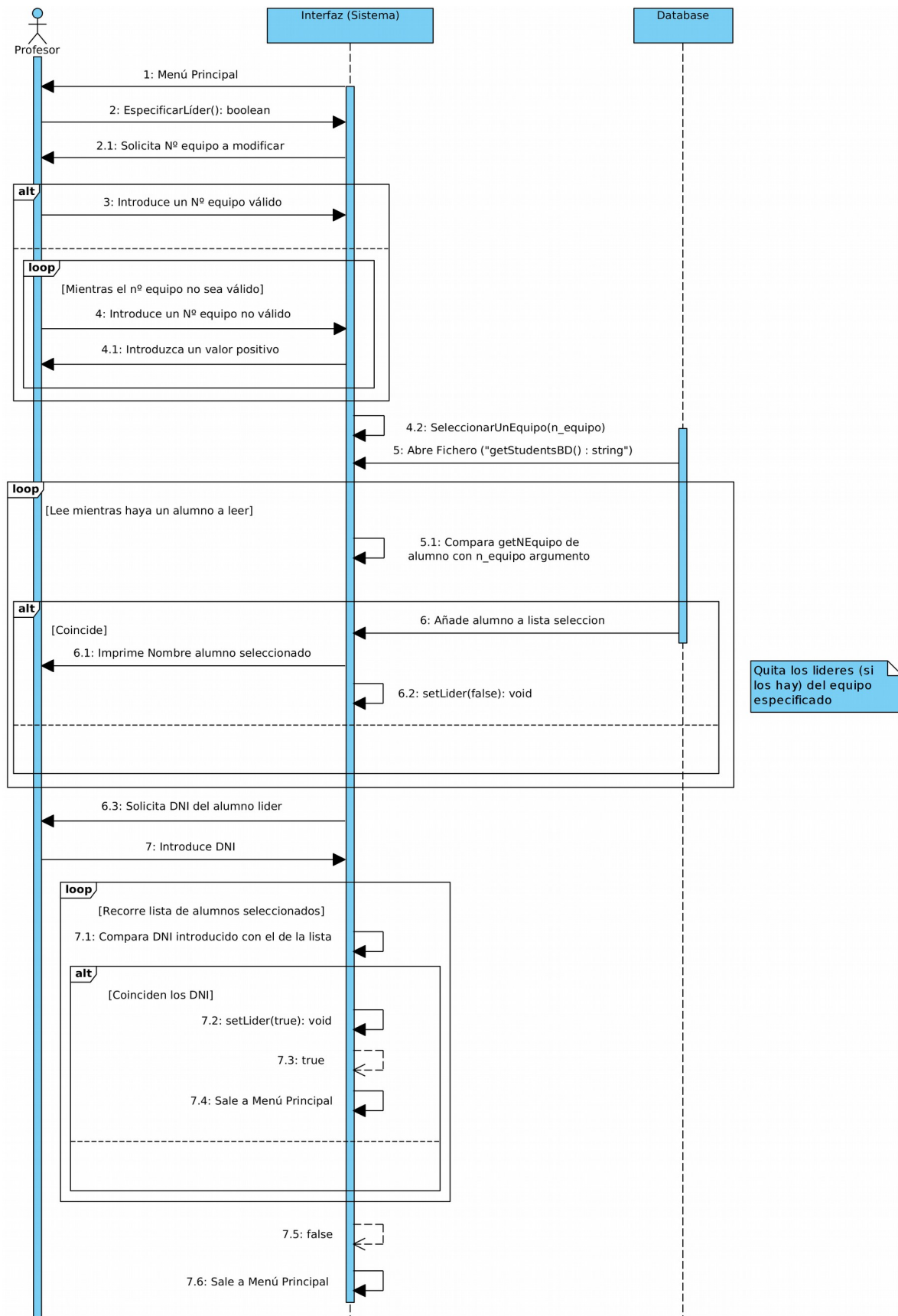
6.3. BORRAR ALUMNO



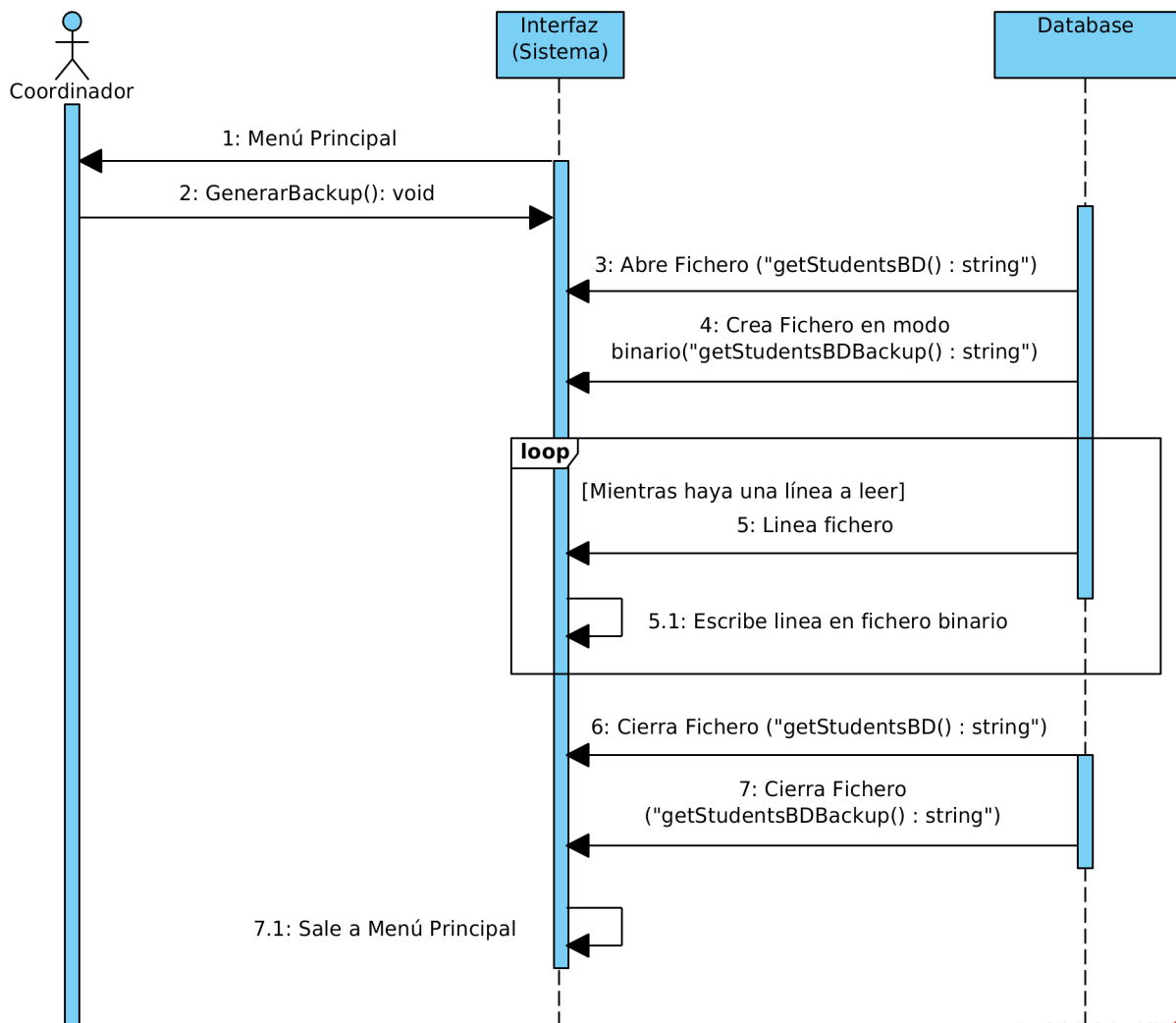
6.4. MOSTRAR ALUMNOS



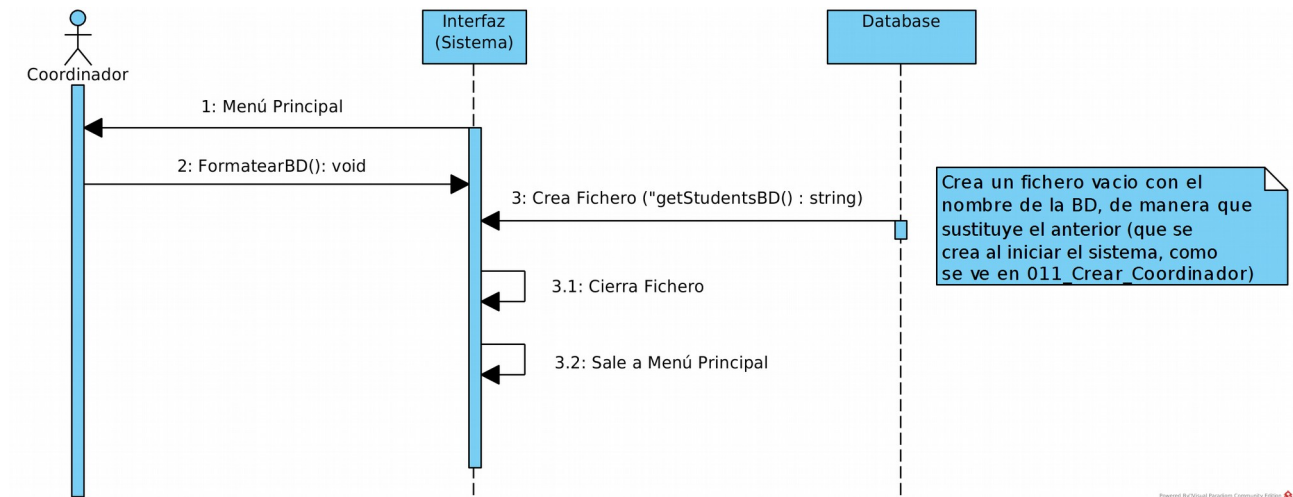
6.5. ESPECIFICAR LÍDER



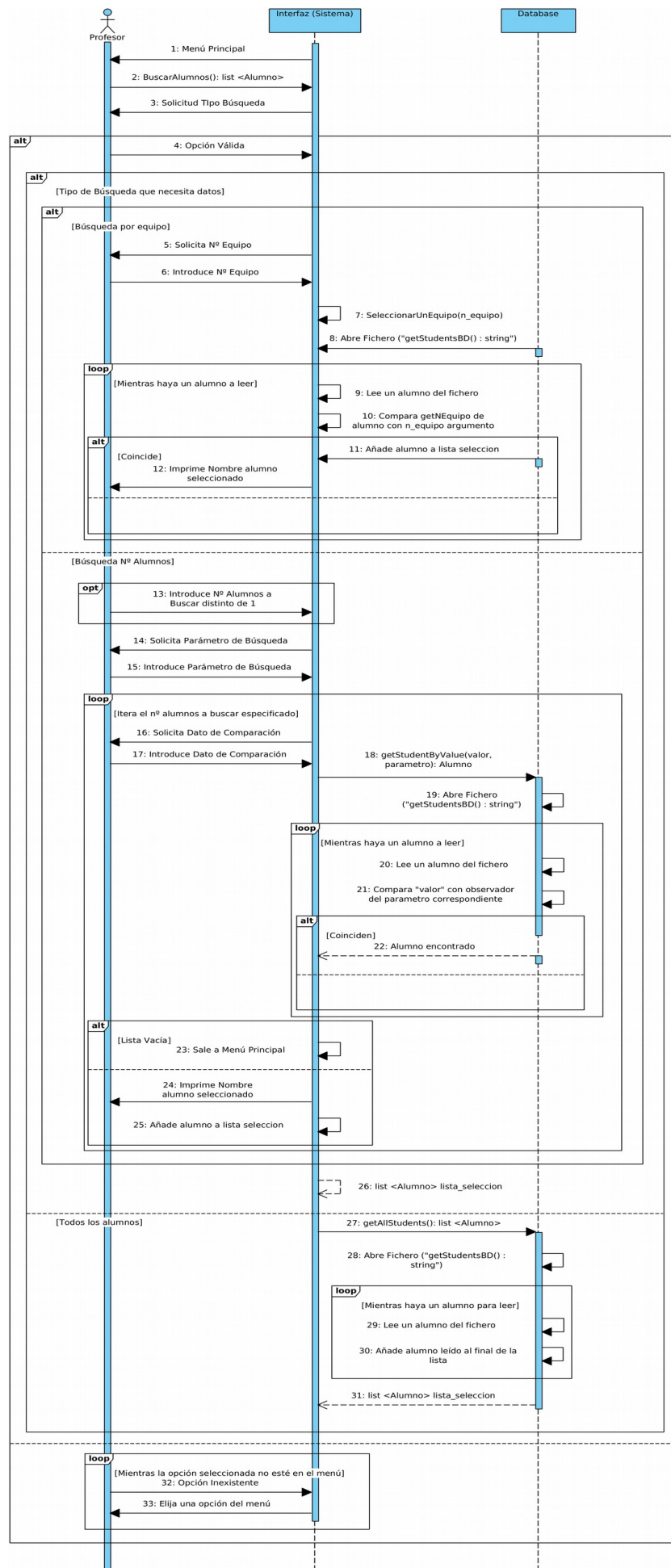
6.6. GENEREAR BACKUP



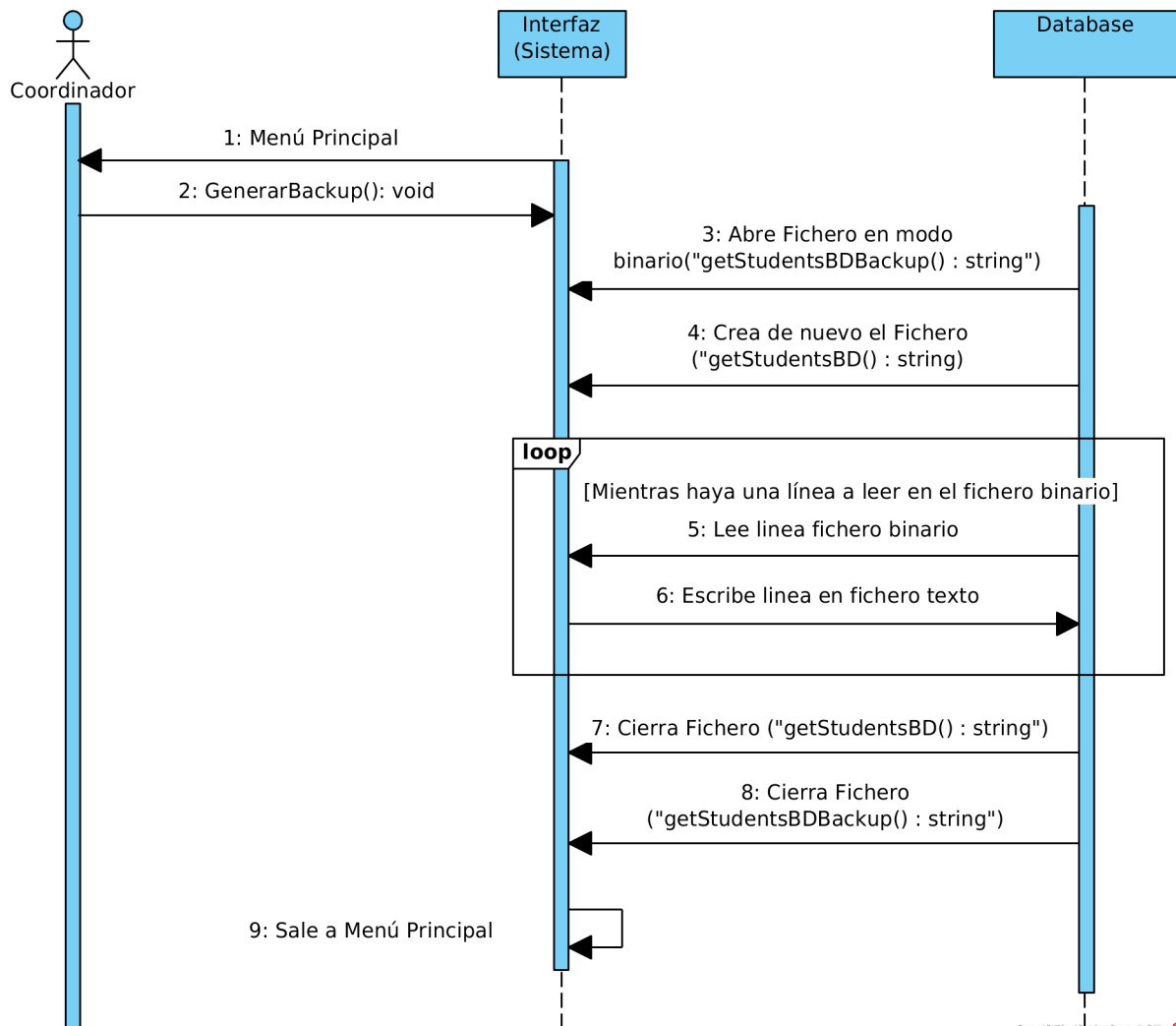
6.7. FORMATEAR BD



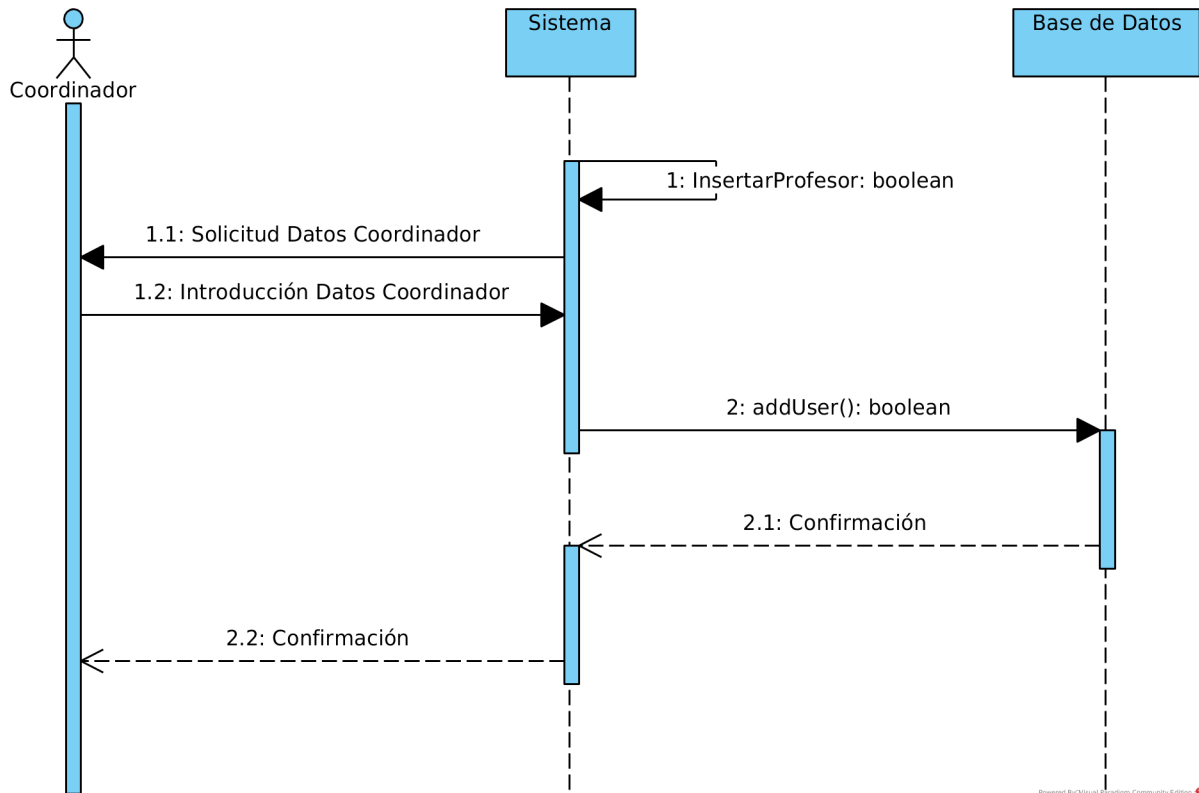
6.8. BUSCAR ALUMNO



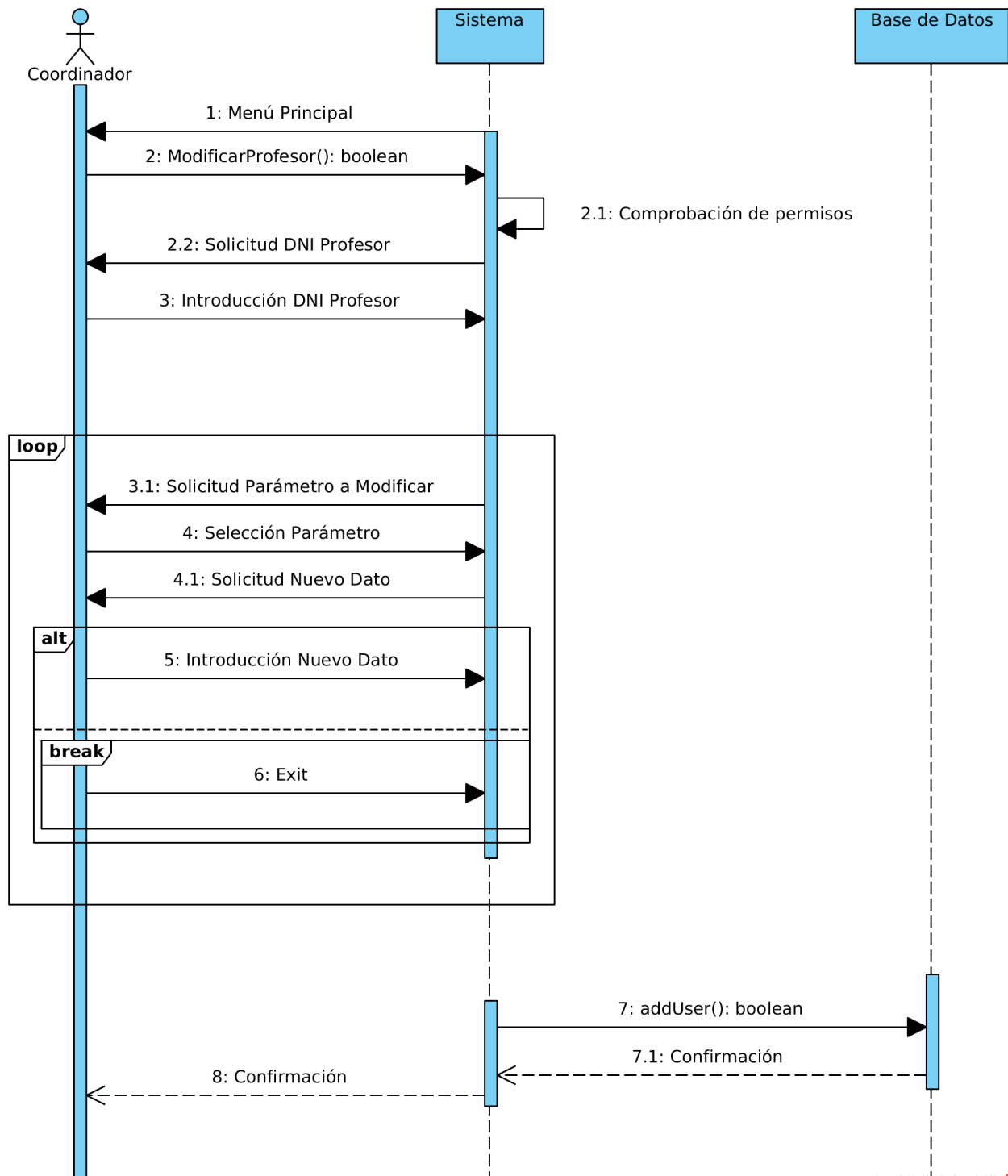
6.9. CARGAR BACKUP



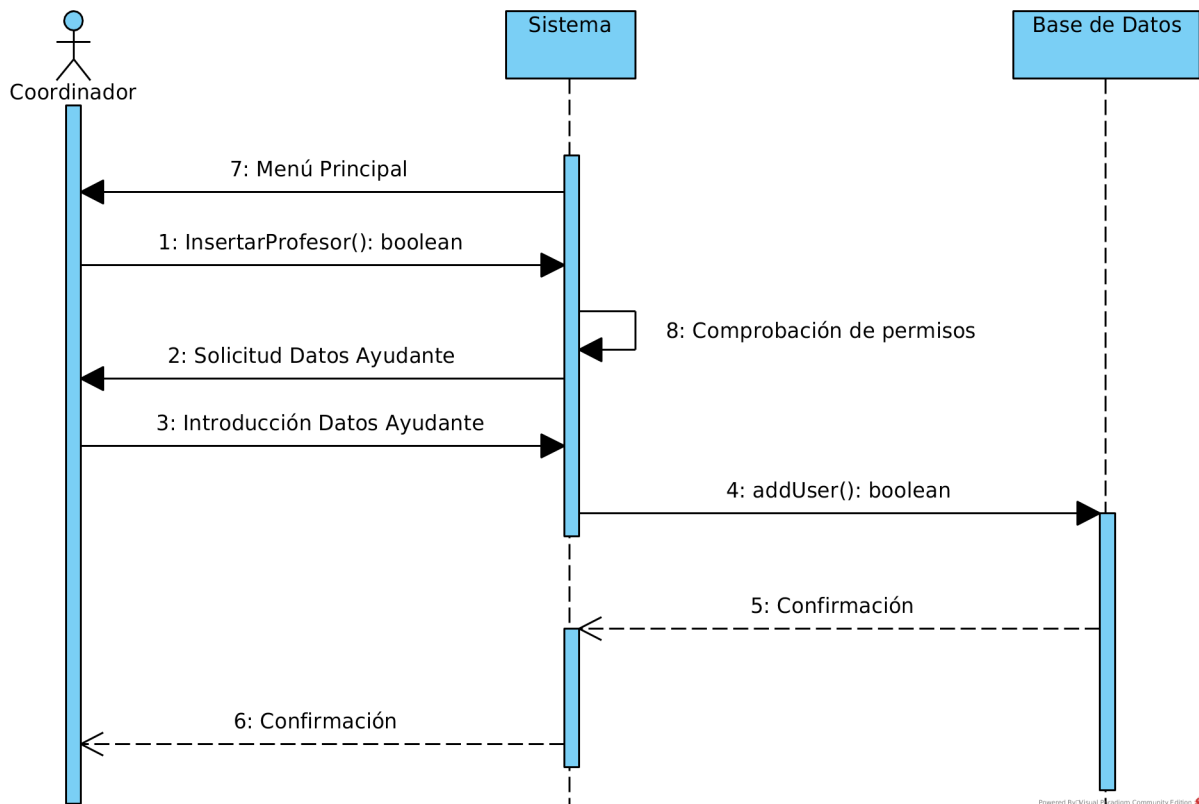
6.10. CREAR COORDINADOR



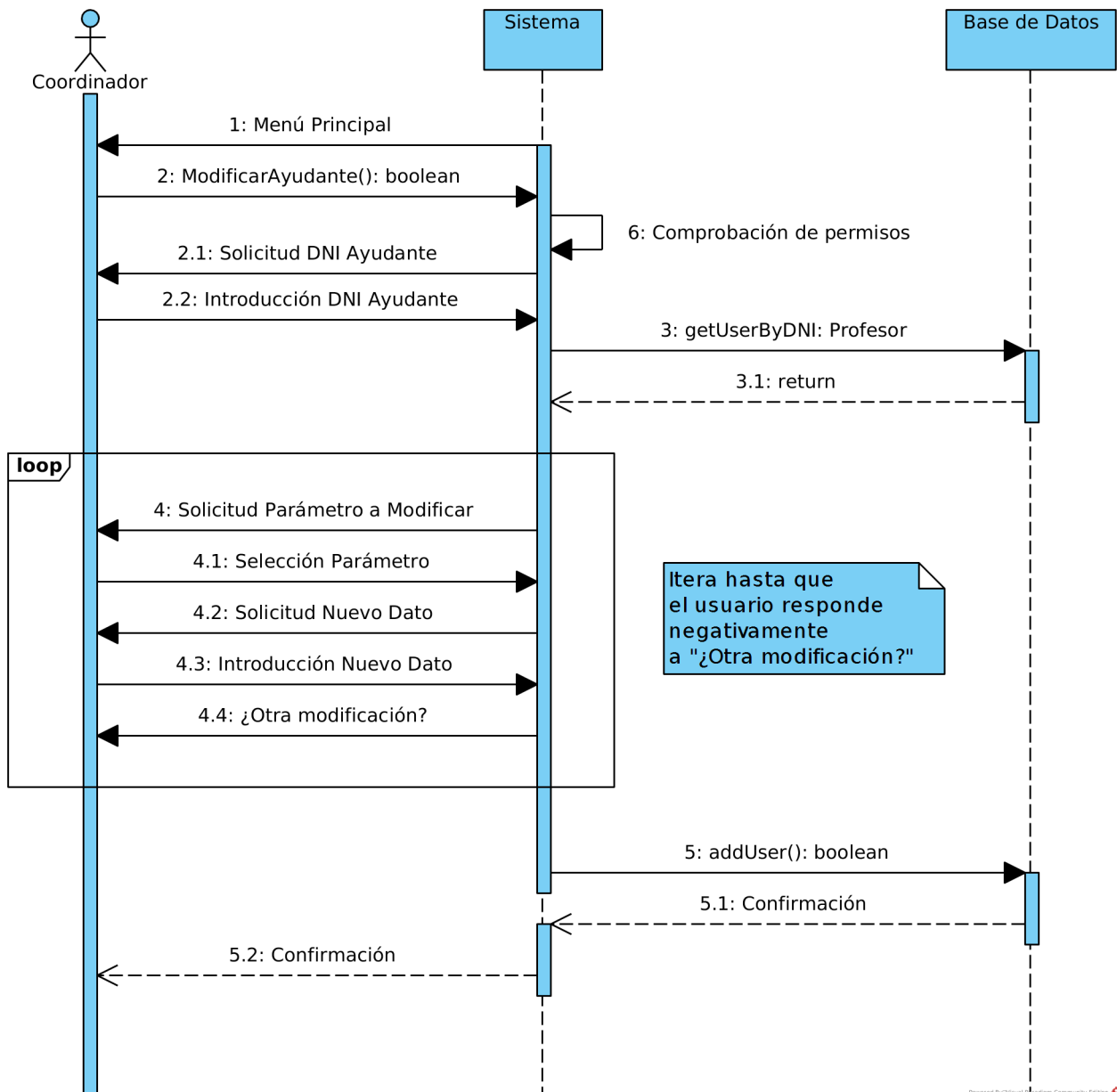
6.11. MODIFICAR PROFESOR



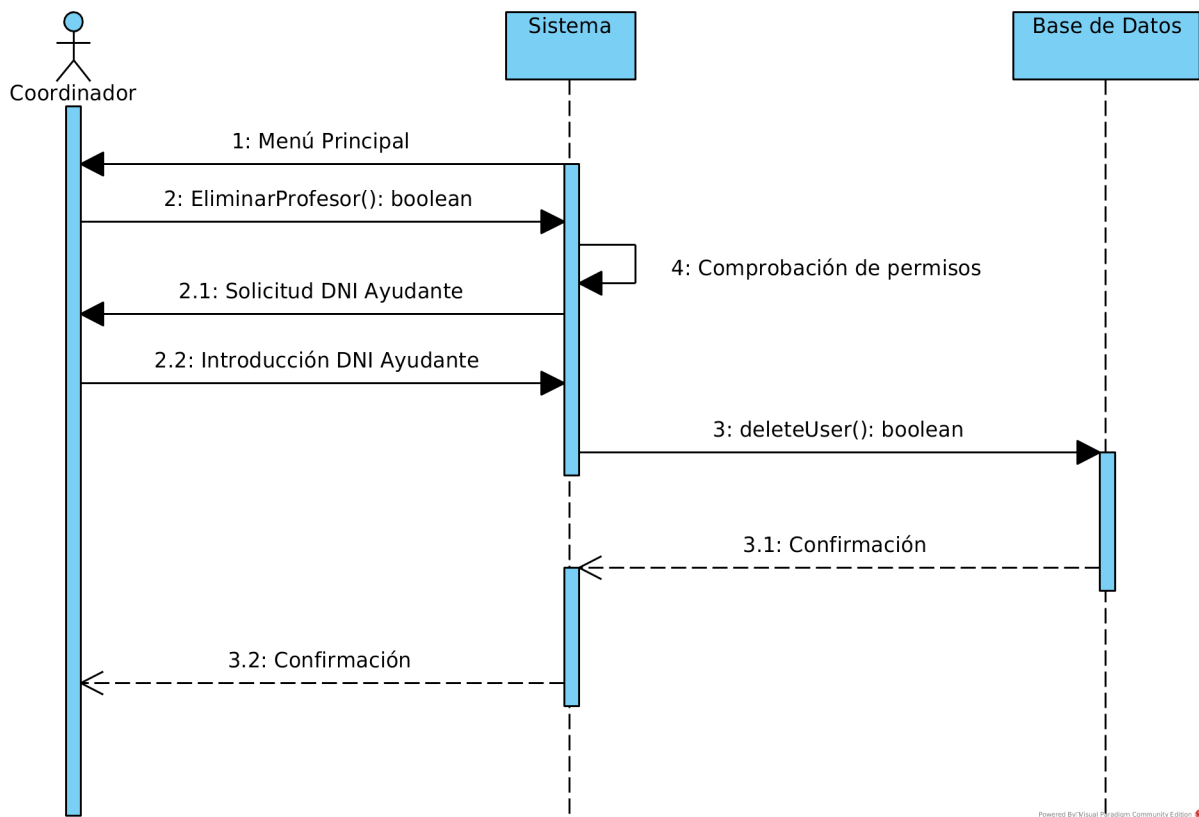
4.12. INSERTAR AYUDANTE



4.13. MODIFICAR AYUDANTE



4.14. ELIMINAR AYUDANTE



7

PRODUCT BACKLOG

1. [Crear Coordinador]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/010_Crear_Coordinador.md) (8 horas estimadas)

2. [Insertar Ayudante]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/012_Insertar_Ayudante.md) (3 horas estimadas)

3. [Insertar Alumno]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/001_Insertar_alumno.md) (5 horas estimadas)

4. [Buscar Alumnos]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/008_Buscar_alumnos.md) (8 horas estimadas)

5. [Modificar Alumno]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/002_Modificar_alumno.md) (2 horas estimadas)

6. [Borrar Alumno]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/003_Borrar_alumno.md) (2 horas estimadas)

7. [Mostrar Alumno]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/004_Mostrar_alumno.md) (2 horas estimadas)

8. [Generar Backup]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/006_Generar_Backup.md) (3 horas estimadas)

9. [Formatear BD]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/007_Formatear_BD.md) (0,5 horas estimadas)

10. [Cargar Backup]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/009_Cargar_Backup.md) (0,5 horas estimadas)

11. [Eliminar Ayudante]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/014_Eliminar_Ayudante.md) (1 hora estimada)

12. [Modificar Coordinador]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/011_Modificar_Coordinador.md) (2 horas estimadas)

13. [Modificar Ayudante]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/013_Modificar_Ayudante.md)
(2 horas estimadas)

14. [Especificar Líder]

(../2_Analisis_de_Requisitos/Historias_de_Usuario/005_Especificar_lider.md) (1 hora estimada)

Total horas estimadas: 40 horas

8

SPRINTS BACKLOG

SPRINTS BACKLOG PRIMERA ENTREGA

Sprint Backlog 1

Manuel Alejandro Luque León

- * Insertar Alumno(1/2)
- * Insertar Ayudante

José Márquez Doblas

- * Crear coordinador(1/2)

Pablo Rodríguez Guillén

- * Buscar Alumnos(1/2)

Sprint Backlog 2

Manuel Alejandro Luque León

- * Modificar Alumno

José Márquez Doblas

- * Crear coordinador(2/2)

Pablo Rodríguez Guillén

- * Borrar Alumno

Sprint Backlog 3

Manuel Alejandro Luque León

- * Mostrar Alumnos

José Márquez Doblas

- * Formatear BD

Pablo Rodríguez Guillén

- * Generar Backup
- * Cargar Backup

SPRINTS BACKLOG SEGUNDA ENTREGA

Sprint Backlog 1

Manuel Alejandro Luque León

* Eliminar ayudante(1/2)

José Márquez Doblas

* Modificar Ayudante

Pablo Rodríguez Guillén

* Buscar Alumnos (2/2)

Sprint Backlog 2

Manuel Alejandro Luque León

* Eliminar Ayudante(2/2)

José Márquez Doblas

* Modificar Coordinador

Pablo Rodríguez Guillén

* Insertar Alumno (2/2)

9

BURNDOWN CHART

