

TRABAJO FINAL DE INGENIERÍA DEL SOFTWARE

Especificación, análisis y diseño del sistema

Informatización de los Alumnos de la Universidad

Autores: Francisco Vera Herencia y Rafael Piqueras Espinar
(i62vehf@uco.es) (i42piesr@uco.es)

**Universidad de Córdoba. Escuela Politécnica Superior de Córdoba.
2o curso, Grado ingeniería en Informática.**



UNIVERSIDAD DE CÓRDOBA



A. LISTA DE FIGURAS

Nombre	Descripción	Página
3.1	HU-1 Buscar Alumno	7
3.2	HU-2 Introducir un nuevo Alumno	8
3.3	HU-3 Cargar Datos	8
3.4	HU-4 Crear Copia de Seguridad	9
3.5	HU-5 Borrar Alumno	9
3.6	HU-6 Modificar Alumno	10
3.7	HU-7 Mostrar Alumno	10
3.8	HU-8 Mostrar Grupo	11
3.9	HU-9 Listar Alumnado completo	11
3.10	HU-10 Log in	12
3.11	HU-11 Registrar Profesor	12
4.1	CU-1 Buscar Alumno	14
4.2	CU-2 Introducir un nuevo Alumno	15
4.3	CU-3 Cargar Datos	16
4.4	CU-4 Crear Copia de Seguridad	17
4.5	CU-5 Borrar Alumno	18
4.6	CU-6 Modificar Alumno	19
4.7	CU-7 Mostrar Alumno	20
4.8	CU-8 Mostrar Grupo	21
4.9	CU-9 Listar Alumnado completo	22
4.10	CU-10 Log in	23
4.11	CU-11 Registrar Profesor	24
4.12	Diagrama de Casos de Uso	25
5.1	Diagrama de Clases	28
5.2	Clase Profesor	29
5.3	Clase Alumno	30
5.4	Clase ListaAlumnos	31
6.1	Diagrama de Secuencia CU-1	33
6.2	Diagrama de Secuencia CU-2	34
6.3	Diagrama de Secuencia CU-3	35

Informatización de Alumnos

6.4	Diagrama de Secuencia CU-4	36
6.5	Diagrama de Secuencia CU-5	37
6.6	Diagrama de Secuencia CU-6	38
6.7	Diagrama de Secuencia CU-7	39
6.8	Diagrama de Secuencia CU-8	40
6.9	Diagrama de Secuencia CU-9	41
6.10	Diagrama de Secuencia CU-10	42
6.11	Diagrama de Secuencia CU-11	43
7.1	Prototipo de Interfaz de menú de acceso	44
7.2	Prototipo de Interfaz de menú principal del coordinador	45
7.3	Prototipo de interfaz de menú principal del ayudante	46
8.1	Sprint Backlog Semana 1	49
8.2	Sprint Backlog Semana 2	49
8.3	Burndown Chart	50

B. LISTAS DE TABLAS

Nombre	Descripción	Página
5.2	Clase Profesor	29
5.3	Clase Alumno	30
5.4	Clase ListaAlumnos	31
9.1	Matriz de Requisitos funcionales/casos de uso	51
9.2	Matriz de Casos de Uso/Clases	52

Índice

Capítulo 1. Definición del Sistema.....	1
1.1. Definición del problema.....	1
1.2. Identificación de los actores del sistema.....	1
Capítulo 2. Extracción de Requisitos.....	3
2.1. Introducción.....	3
2.2. Requisitos funcionales.....	2
2.3. Requisitos no funcionales.....	3
Capítulo 3. Historias de Usuario.....	4
Capítulo 4. Casos de Uso.....	13
4.1. Casos de Uso.....	14
4.2. Diagrama de Casos de Uso.....	25
Capítulo 5. Diagrama de Clases.....	27
5.1. Definición de las clases del sistema.....	28
Capítulo 6. Diagramas de Secuencia.....	33
Capítulo 7. Diseño de la Interfaz.....	44
7.1. Descripción de la Interfaz.....	44
Capítulo 8. Metodología SCRUM.....	47
8.1. Product Backlog.....	47
8.2. Sprint Backlogs.....	49
8.3. BurnDown Chart.....	50
Capítulo 9. Técnicas de Validación.....	51
9.1. Matriz RF\CU.....	51
9.2. Matriz CU\Clases.....	52

Capítulo 1. Definición del sistema.

1.1. Definición del problema.

El presente software viene a resolver el problema que nos plantea los diferentes profesores que imparten la asignatura de ingeniería de software para informatizar los datos de contacto de todos los alumnos.

El cliente nos ha redefinido el problema pidiéndonos considerar la posibilidad de registrar diferentes profesores para que solo puedan acceder al sistema aquellos cuyas credenciales se encuentren registradas.

Para ello el cliente nos pide que contemplemos el hecho de que los profesores pueden tener dos roles, bien coordinador o bien ayudante. Asimismo, el cliente nos pide que tengamos en cuenta que las tareas son las mismas para ambos roles, a excepción del trabajo con las copias de seguridad que serán exclusivas del coordinador de la asignatura.

1.2. Identificación de los actores del sistema.

A continuación, se describen los tipos de actores que interactúan con el presente sistema en distintos grados, junto con el papel que van a realizar con respecto al software.

Profesor: Se trata del actor principal, ya que será quien acceda al sistema y es el encargado de manejar la Agenda de Alumnos puesto que hace todas las funciones que tiene el sistema desde introducir un alumno hasta modificar y eliminar, etc.

Alumno: Este actor no interactúa de forma directa en el sistema, aunque sin él no existiría la Agenda de Alumnos. Por tanto, se trata de un actor pasivo, no por ello menos importante en la estructura del sistema, ya que es el elemento base.

Capítulo 2. Extracción de Requisitos.

2.1. Introducción.

Mediante la redacción del siguiente capítulo de la documentación del sistema, se pretende elicitar los requisitos tanto funcionales como no funcionales, con el objetivo de facilitar la comprensión e instalación de este sistema.

Para ello no se va a emplear ningún estándar en particular. Los requisitos se han extraído de los objetivos indicados por el cliente, así como de una entrevista realizada previamente con él. Para mayor comprensión de los requisitos se usará una notación anidada para mayor especialización y perfección.

Los siguientes requisitos hacen referencia a las exigencias que debe cumplir el sistema en sí para el correcto funcionamiento del presente software.

2.2 Requisitos Funcionales.

Se muestran a continuación los requisitos de funcionamiento del sistema:

RF-1 El sistema debe ser capaz de buscar un alumno entre los ya registrados mediante su DNI, sus apellidos o su grupo.

RF-2 Introducción de nuevos alumnos de forma manual. Durante el registro se deberán introducir todos los datos del alumno a excepción del equipo y si es líder o no del mismo.

RF-3 Se podrá cargar desde un fichero o copia de seguridad un conjunto de alumnos.

RF-4 El sistema contará con una opción para generar una nueva copia de seguridad cuando el usuario/cliente lo desee.

RF-5 Borrar un alumno.

RF-6 Modificar los datos de un alumno.

RF-7 El sistema podrá mostrar un alumno en concreto.

RF-8 El sistema tendrá la opción de mostrar los alumnos de un único grupo en concreto.

RF-9 El sistema contará con la opción de mostrar de forma ordenada ascendente o descendentemente el conjunto de todos los alumnos registrados.

RF-10 Los usuarios podrán acceder al sistema iniciando sesión con su identificador y contraseña únicos.

RF-11 El sistema permitirá al profesor coordinador que registre a nuevos profesores.

2.2.1 Prioridad de Requisitos Funcionales.

RF-1 -> Prioridad 1

RF-2 -> Prioridad 2

RF-3 -> Prioridad 3

RF-4 -> Prioridad 3

RF-5 -> Prioridad 4

RF-6 -> Prioridad 5

RF-7 -> Prioridad 5

RF-8 -> Prioridad 5

RF-9 -> Prioridad 5

RF-10 -> Prioridad 5

RF-11 -> Prioridad 5

2.3 Requisitos no Funcionales.

RNF-1 La interfaz utilizada será la línea de comandos.

RNF-2 El lenguaje de programación utilizado para este sistema será C++.

RNF-3 El lenguaje de documentación utilizado en este sistema será Markdown.

RNF-4 Las copias de seguridad serán guardadas en ficheros binarios, tanto los admitidos como entrada como los generados por el sistema.

RNF-5 El número máximo de alumnos que soportará el sistema será de 150 alumnos.

RNF-6 El sistema será operativo en SOs Linux.

RNF-7 A la hora de mostrar un alumno se mostrará mediante la interfaz.

RNF-8 Los profesores tendrán un rol de entre dos: coordinador y ayudante.

RNF-9 Los profesores ayudantes no podrán crear ni cargar copias de seguridad, así como registrar a nuevos profesores.

RNF-11 Los nuevos profesores registrados serán ayudantes por defecto.

Capítulo 3. Historias de Usuario

Historia de usuario 1 se corresponde con el requisito funcional 1, Buscar alumno.

(Anverso)

ID: 001 Buscar alumno

Como usuario quiero poder buscar un alumno.

Prioridad: 1

(Reverso)

- Quiero poder buscar al alumno por su DNI, apellido o grupo.

Figura 3.1 HU-1 Buscar alumno

Historia de usuario 2 se corresponde con el requisito funcional 2, introducir un nuevo alumno.

(Anverso)

ID: 002 Introducir un nuevo alumno

Como usuario quiero poder introducir un nuevo alumno.

Prioridad: 2

(Reverso)

- Como usuario quiero poder introducir todos los datos del alumno a excepción del grupo y su líder.
- El sistema comprobará si el alumno está registrado.

Figura 3.2 HU-2 Introducir un nuevo alumno

Historia de usuario 3 se corresponde con el requisito funcional 3, Cargar datos.

(Anverso)

ID: 003 Cargar datos

Como usuario quiero poder cargar una lista de alumnos.

Prioridad: 3

(Reverso)

- Como usuario quiero poder cargar esa lista de alumnos desde un fichero o copia de seguridad.

Figura 3.3 HU-3 Cargar datos

Historia de usuario 4 se corresponde con el requisito funcional 4, Crear copia de seguridad.

(Anverso)

ID: 004 Crear copia de seguridad

Como usuario quiero poder crear una copia de seguridad.

Prioridad: 3

(Reverso)

- Como usuario quiero hacer una copia de seguridad en cualquier momento.
- Se debe crear una nueva copia de seguridad sin necesidad de sobrescribir la anterior.
- Solo se sobrescribirá en caso de querer generar varias el mismo día.

Figura 3.4 HU-4 Crear copia de seguridad

Historia de usuario 5 se corresponde con el requisito funcional 5, Borrar alumno.

(Anverso)

ID: 005 Borrar alumno

El sistema debe poder borrar un alumno.

Prioridad: 4

(Reverso)

- Se debe comprobar que el alumno se encuentra registrado.
- Deben quedar borrados todos los datos del alumno.
- Si el alumno a borrar era líder de su grupo, el grupo quedará sin líder hasta modificación de los datos de alguno de los miembros restantes.

Figura 3.5 HU-5 Borrar alumno

Historia de usuario 6 se corresponde con el requisito funcional 6, Modificar alumno.

(Anverso)

ID: 006 Modificar alumno

El sistema debe poder modificar los datos de un alumno.

Prioridad: 5

(Reverso)

- Se deben mostrar todos los datos del alumno antes y después de la modificación.
- Se debe poder cambiar cualquier dato del alumno.
- El sistema debe comprobar que el alumno existe.

Figura 3.6 HU-6 Modificar alumno

Historia de usuario 7 se corresponde con el requisito funcional 7, Mostrar alumno.

(Anverso)

ID: 007 Mostrar alumno

El sistema debe poder mostrar los datos de un alumno.

Prioridad: 5

(Reverso)

- Se deben mostrar todos los datos de un alumno en concreto.
- El sistema debe comprobar que el alumno existe.

Figura 3.7 HU-7 Mostrar alumno

Historia de usuario 8 se corresponde con el requisito funcional 8, Mostrar grupo.

(Anverso)

ID: 008 Mostrar grupo

El sistema debe poder mostrar los alumnos que forman un grupo de trabajo.

Prioridad: 5

(Reverso)

- El sistema mostrará los nombres y apellidos de los componentes del grupo.
- Para encontrar el grupo y sus alumnos se usará el ID del grupo.
- Se debe comprobar que el grupo existe.

Figura 3.8 HU-8 Mostrar grupo

Historia de usuario 9 se corresponde con el requisito funcional 9, Listar alumnado completo.

(Anverso)

ID: 009 Listar alumnado completo

El sistema debe poder mostrar una lista de todos los alumnos registrados.

Prioridad: 5

(Reverso)

- Se mostrarán de forma ordenada creciente o decrecientemente.
- El orden será por apellidos o grupo.

Figura 3.9 HU-9 Listar alumnado completo

Historia de usuario 10 se corresponde con el requisito funcional 10, Log in.

(Anverso)

ID: 010 Log in

Los usuario podrán acceder al sistema iniciando sesión en el mismo.

Prioridad: 5

(Reverso)

- Para el inicio de sesión será necesario el ID y la contraseña.
- Ningún usuario tendrá acceso al sistema si no está registrado.

Figura 3.10 HU-10 Log in

Historia de usuario 11 se corresponde con el requisito funcional 11, Registrar profesor.

(Anverso)

ID: 011 Registrar profesor

Se desea crear un nuevo perfil de profesor.

Prioridad: 5

(Reverso)

- El profesor creado será ayudante, por defecto.
- Solo podrá acceder a esta sección el coordinador.
- Al menos, se introducirán los datos correspondientes al Nombre e ID del nuevo profesor.

Figura 3.11 HU-11 Registrar profesor

Capítulo 4. Casos de Uso.

En este capítulo se expondrán los diferentes casos de uso presentes en el sistema, así como el correspondiente diagrama de casos de uso.

4.1. Casos de Uso.

Caso de uso 1 se corresponde con la historia de usuario 1, Buscar alumno.

Buscar alumno

ID: 001

Breve descripción: El sistema debe poder buscar un alumno

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El alumno debe estar registrado.

Flujo principal:

1. El sistema necesita buscar un alumno.
2. Se necesita un criterio de búsqueda: DNI, apellidos o grupo.
3. Se filtra en la lista de alumnos para encontrarlo.

Postcondiciones:

1. El sistema recoge los datos del alumno

Flujos alternativos:

- 3a. Si el alumno no se encuentra, se muestra un mensaje de error y vuelve al menú principal.
- 3b. Se recogerán los datos de todos los alumnos que coincidan con el criterio de búsqueda, en caso de que ocurra.

Figura 4.1 CU-1 Buscar alumno

Caso de uso 2 se corresponde con la historia de usuario 2, Introducir un nuevo alumno.

Introducir un nuevo alumno

ID: 002

Breve descripción: El sistema debe poder introducir un nuevo alumno

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El alumno no debe estar registrado.
2. No debe haber 150 alumnos registrados.

Flujo principal:

1. El sistema necesita introducir un nuevo alumno.
2. El sistema pide los datos del nuevo alumno.
3. El sistema recibe los datos del nuevo alumno a excepción del grupo y su líder.
4. Se comprueba si el alumno no está registrado.
5. Se guardan los datos del alumno.

Postcondiciones:

1. El nuevo alumno queda registrado en el sistema.

Flujos alternativos:

- 3a. Si los datos introducidos no son válidos, se muestra un mensaje de error y se vuelven a pedir los datos.
- 4a. Si el alumno ya está registrado se muestra un mensaje de error y vuelve al menú principal.

Figura 4.2 CU-2 Introducir un nuevo alumno

Casos de uso 3 se corresponde con historia de usuario 3, Cargar datos.

Cargar Datos

ID: 003

Breve descripción: El sistema debe poder cargar una lista de alumnos.

Actores principales: Profesor Coordinador

Actores secundarios: Alumno

Precondiciones:

1. El sistema debe tener una lista de alumnos en un fichero o copia de seguridad.
2. El usuario debe ser coordinador.

Flujo principal:

1. El sistema necesita cargar de un fichero datos sobre alumnos.
2. El sistema pregunta cual es el fichero o copia a cargar.
3. Se carga la lista de alumnos en el sistema desde el fichero indicado.
4. El sistema mostrará un mensaje de que la lista se cargó correctamente.

Postcondiciones:

1. El sistema debe tener guardados todos los datos de la lista de alumnos que hubiese en la copia de seguridad.

Flujo alternativo:

- 2a. Si no se encuentra el fichero, se muestra un mensaje de error.
- 2b. Se vuelve al menú principal.
- 3a. El sistema mostrará un mensaje de error si la lista no se cargó correctamente.
- 3b. Se vuelve al menú principal.

Figura 4.3 CU-3 Cargar datos

Casos de uso 4 se corresponde con la historia de usuario 4, Crear copia de seguridad.

Crear Copia de Seguridad

ID: 004

Breve descripción: El sistema debe crear una copia de seguridad

Actores principales: Profesor Coordinador

Actores secundarios: Alumno

Precondiciones:

1. El sistema debe tener datos de alumnos guardados.
2. El usuario debe ser coordinador.

Flujo principal:

1. El sistema necesita crear una copia de seguridad.
2. El sistema creará una copia de seguridad sin necesidad de sobrescribir la anterior.
3. Se muestra un mensaje de que se ha grabado el fichero correctamente.

Postcondiciones:

1. El sistema crea la copia de seguridad.

Flujo alternativo:

- 2a. El sistema mostrará un mensaje de error si no se creó la copia de seguridad y vuelve al menú principal.

Figura 4.4 CU-4 Crear copia de seguridad

Caso de uso 5 se corresponde con la historia de usuario 5, Borrar alumno.

Borrar Alumno

ID: 005

Breve Descripción: El sistema borrará un alumno

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El alumno debe estar registrado.

Flujo principal:

1. El sistema necesita borrar un alumno.
2. Se pide el DNI o los apellidos del alumno.
3. El sistema busca el alumno.
4. Se borran los datos del alumno.
5. El sistema mostrará un mensaje confirmando que el alumno ha sido borrado.

Postcondiciones:

1. El sistema borrará todos los datos del alumno.

Flujo alternativo:

3a. Si no encuentra el alumno, muestra un mensaje de error y vuelve al paso anterior (pedir datos).

3b. Si, al buscar por apellidos, se encuentran varios alumnos, se mostrarán sus datos (apellidos, nombre y DNI). Después se pedirá al usuario que elija entre los encontrados para borrarlo.

5a. Si no se eliminan correctamente los datos, se muestra un mensaje de error y vuelve al menú principal.

Figura 4.5 CU-5 Borrar alumno

Caso de uso 6 se corresponde con la historia de usuario 6, Modificar alumno.

Modificar Alumno

ID: 006

Breve descripción: El sistema puede modificar los datos de un alumno.

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El alumno debe estar registrado

Flujo principal:

1. El sistema necesita modificar los datos de un alumno.
2. Se piden los datos del alumno para buscarlo: DNI o apellidos.
3. Se busca el alumno.
4. El sistema pregunta cuál es el dato a modificar.
5. Se pide el nuevo valor.

Postcondiciones:

1. El sistema modifica los datos del alumno.

Flujos alternativos:

3a. Si el alumno no se encuentra, se muestra un mensaje de error.

3b. Si, al buscar por apellidos, se encuentran varios alumnos, se mostrarán sus datos (apellidos, nombre y DNI). Después se pedirá elegir cual de los alumnos encontrados es el buscado.

Figura 4.6 CU-6 Modificar alumno

Caso de uso 7 se corresponde con la historia de usuario 7, Mostrar alumno.

Mostrar alumno

ID: 007

Breve descripción: El sistema debe poder mostrar los datos de un alumno.

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El alumno debe estar registrado en el sistema.

Flujo principal:

1. El sistema necesita mostrar un alumno.
2. El sistema pide el criterio de búsqueda al usuario, que debe ser DNI o apellidos.
3. El sistema busca el alumno.
4. Se recogen los datos del alumno.

Postcondiciones:

1. Se muestran los datos del alumno.

Flujos alternativos:

- 2a. En caso de que se introduzca un DNI, si no es correcto, se muestra un mensaje de error y da la opción de introducir un nuevo criterio de búsqueda.
3. Si el alumno no existe muestra un mensaje de error y vuelve al menú principal.

Figura 4.7 CU-7 Mostrar alumno

Caso de uso 8 se corresponde con la historia de usuario 8, Mostrar grupo.

Mostrar grupo

ID: 008

Breve descripción: El sistema debe poder mostrar los alumnos que forman un grupo de trabajo.

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El grupo debe estar registrado

Flujo principal:

1. El sistema necesita mostrar los alumnos de un grupo.
2. El sistema pide al usuario el ID del grupo.
3. El sistema busca los alumnos por la ID de su grupo.
4. Se muestran los datos de los componentes, resaltando al líder.

Postcondiciones:

1. Se muestran los datos de los componentes del grupo.

Flujos alternativos:

- 2a. Si se introduce un ID que no es válido, se muestra un mensaje de error y permite volver a introducir el ID.
- 3a. Si no se encuentra ningún alumno con dicho grupo asignado se muestra un mensaje de error y vuelve al menú principal.

Figura 4.8 CU-8 Mostrar grupo

Caso de uso 9 se corresponde con la historia de usuario 9, Listar alumnado completo.

Listar alumnado completo

ID: 009

Breve descripción: El sistema debe poder mostrar una lista de todos los alumnos registrados.

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. Debe haber alumnos registrados.

Flujo principal:

1. El sistema necesita listar los alumnos.
2. Se ordenan los alumnos internamente según el criterio establecido.
3. El sistema genera una lista de alumnos que se imprimirá por pantalla.

Postcondiciones:

1. El sistema muestra la lista de todos los alumnos ordenados.

Flujos alternativos:

- 3a. Si ocurre cualquier problema se imprime un mensaje de error y vuelve al menú.

Figura 4.9 CU-9 Listar alumnado completo

Caso de uso 10 se corresponde con la historia de usuario 10, Log in.

Log in

ID: 010

Breve descripción: Inicio de sesión para usuarios.

Actores principales: Profesor

Actores secundarios: Alumno

Precondiciones:

1. El usuario debe estar registrado en el sistema.

Flujo principal:

1. El usuario quiere iniciar sesión.
2. El sistema solicita ID y contraseña.
3. El sistema comprueba si el usuario está registrado.
4. Se validan los datos del usuario.
5. El usuario accede al sistema.

Postcondiciones:

1. El sistema permite al usuario acceder a todas las opciones a las que según su rol tenga permiso.

Flujos alternativos:

- 3a. Si el usuario no está registrado, se muestra un mensaje de error y termina la ejecución del programa.
- 4a. Si los datos introducidos no son correctos, se muestra un mensaje de error y termina la ejecución del programa.

Figura 4.10 CU-10 Log in

Caso de uso 11 se corresponde con la historia de usuario 11, Registrar profesor.

Registrar profesor

ID: 011

Breve descripción: Se desea crear un nuevo perfil de profesor.

Actores principales: Profesor Coordinador

Actores secundarios: Profesor

Precondiciones:

1. El usuario debe haber iniciado sesión.
2. El usuario debe ser coordinador.

Flujo Principal:

1. El sistema necesita registrar un nuevo profesor.
2. Se piden los datos del nuevo profesor.
3. El sistema comprueba si ya estaba registrado.
4. El sistema guarda los datos del nuevo profesor.
5. Se imprime un mensaje de éxito.

Postcondiciones:

1. El sistema tiene un nuevo profesor.

Flujo alternativo:

- 3a. Si ya estaba registrado se imprime un mensaje de error y vuelve al menú.

Figura 4.11 CU-11 Registrar profesor

4.2. Diagrama de Casos de Uso.

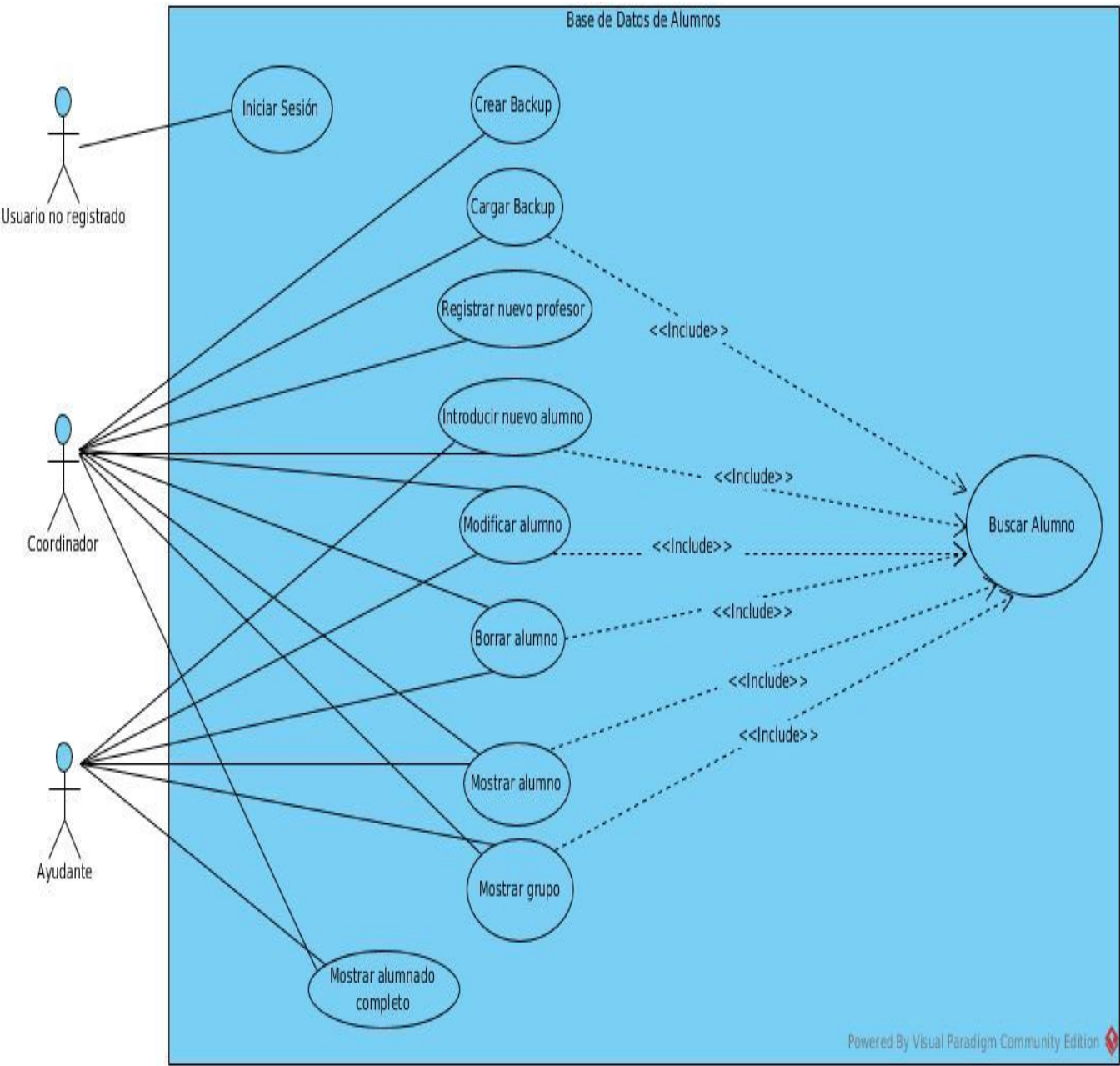


Figura 4.12 Diagrama de Casos de Uso

Capítulo 5. Diagrama de clases.

En este capítulo se presentan tanto el diagrama de clases como las descripciones detalladas de cada una de las clases que componen el sistema.

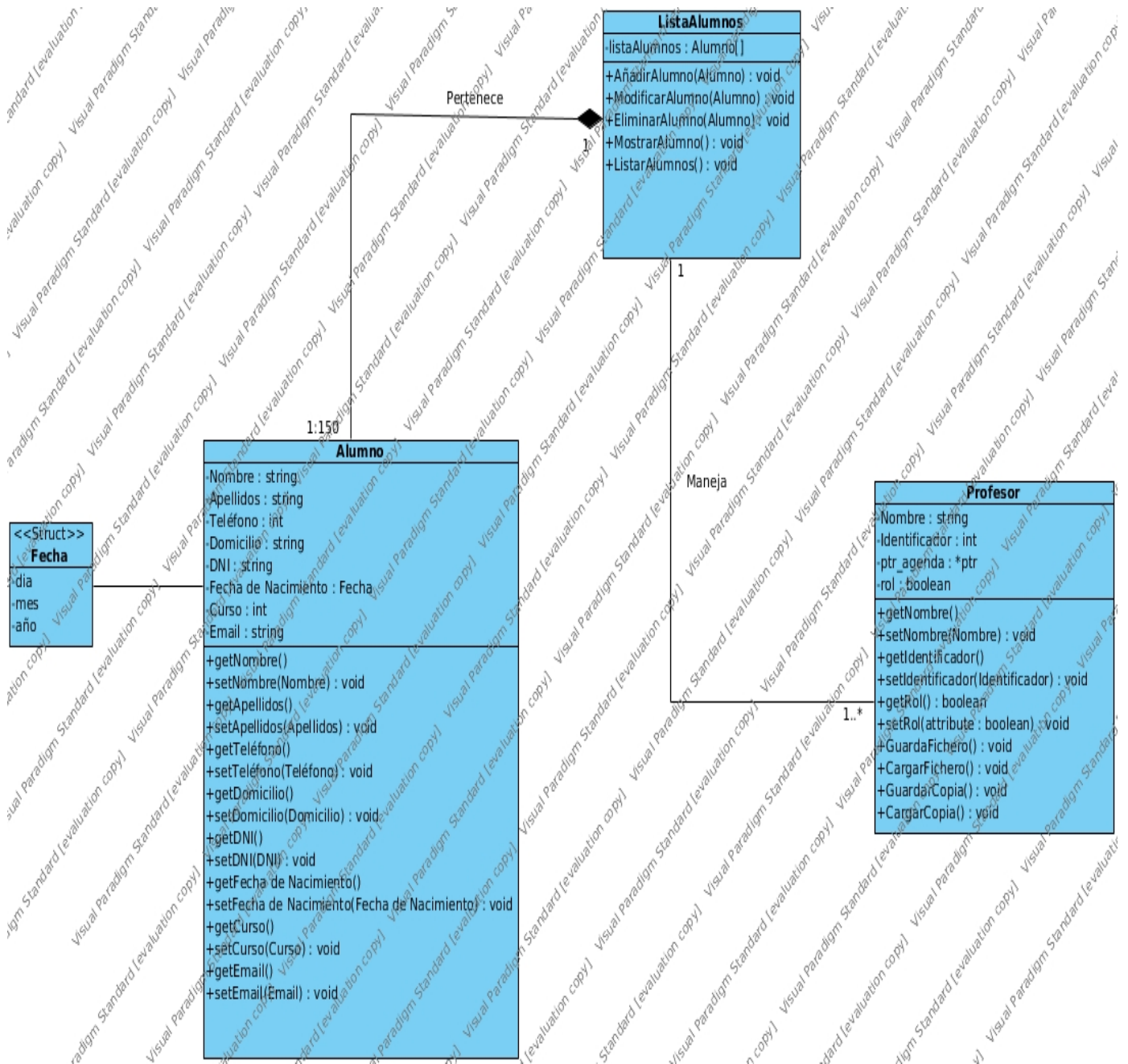


Figura 5.1 Diagrama de Clases

5.1 Descripción de las clases del sistema.

A partir de la información expuesta en los apartados anteriores se procede a la descripción de las clases conformantes del sistema de gestión de los datos de los alumnos y de los profesores registrados.

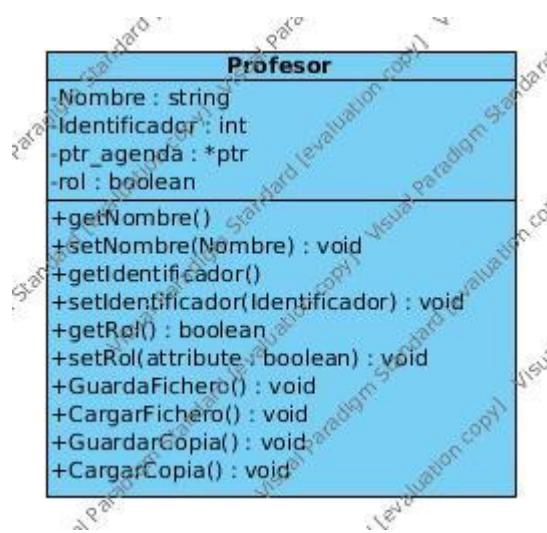


Figura 5.2 Clase Profesor

Clase	Profesor
Descripción	En la clase Profesor se procesan los datos correspondientes a sus atributos para poder acceder al sistema y realizar las operaciones que se detallan.
Atributos	<ul style="list-style-type: none"> • Nombre • Identificador • agenda • rol
Operaciones	<ul style="list-style-type: none"> • getNombre() • setNombre(Nombre) • getIdentificador() • setIdentificador(Identificador) • getRol() • setRol() • GuardarFichero() • CargarFichero() • GuardarCopia() • CargarCopia()

Tabla 5.2 Clase Profesor

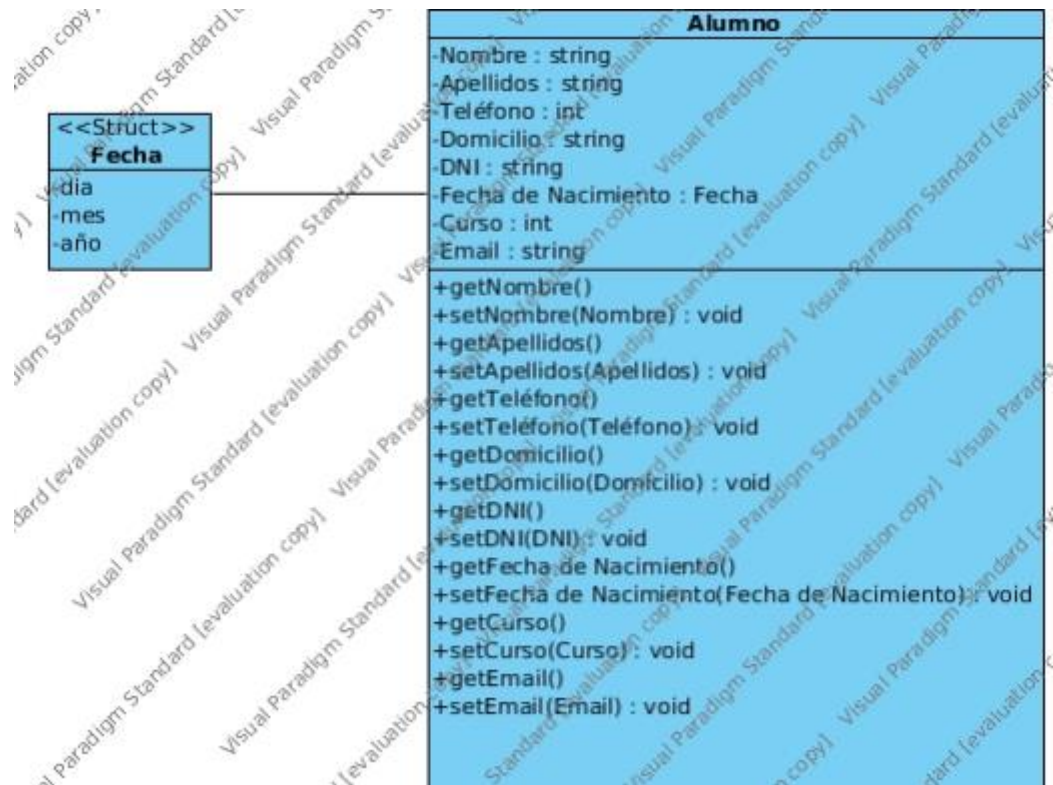


Figura 5.3 Clase Alumno

Clase	Alumno
Descripción	En la clase Alumno se procesan los datos principales como a continuación se pueden observar en los atributos.
Atributos	<ul style="list-style-type: none"> • Nombre • Apellidos • Teléfono • Domicilio • DNI • Fecha de nacimiento • Curso • Email
Operaciones	<ul style="list-style-type: none"> • getNombre() • setNombre(Nombre) • getApellidos() • setApellidos(Apellidos) • getTelefono() • setTelefono(Telefono) • getDomicilio() • setDomicilio(Domicilio) • getDNI() • setDNI(DNI) • getFechaNacimiento(FechaNacimiento) • getCurso() • setCurso(curso) • getEmail() • setEmail(Email)

Tabla 5.3 Clase Alumno



Figura 5.4 Clase ListaAlumnos

Clase	ListaAlumnos
Descripción	En la clase ListaAlumnos se procesan los datos correspondientes a una lista de alumnos y se pueden realizar las operaciones que se detallan en las operaciones.
Atributos	<ul style="list-style-type: none"> • ListaAlumnos Alumno()
Operaciones	<ul style="list-style-type: none"> • AñadirAlumno(Alumno) • ModificarAlumno(Alumno) • EliminarAlumno(Alumno) • MostrarAlumno() • ListarAlumnos()

Tabla 5.4 Clase ListaAlumnos

Capítulo 6. Diagramas de Secuencia

A continuación, se muestran los diagramas de secuencia de los casos de uso presentes en el sistema.

- **Diagrama de secuencia CU-1.**

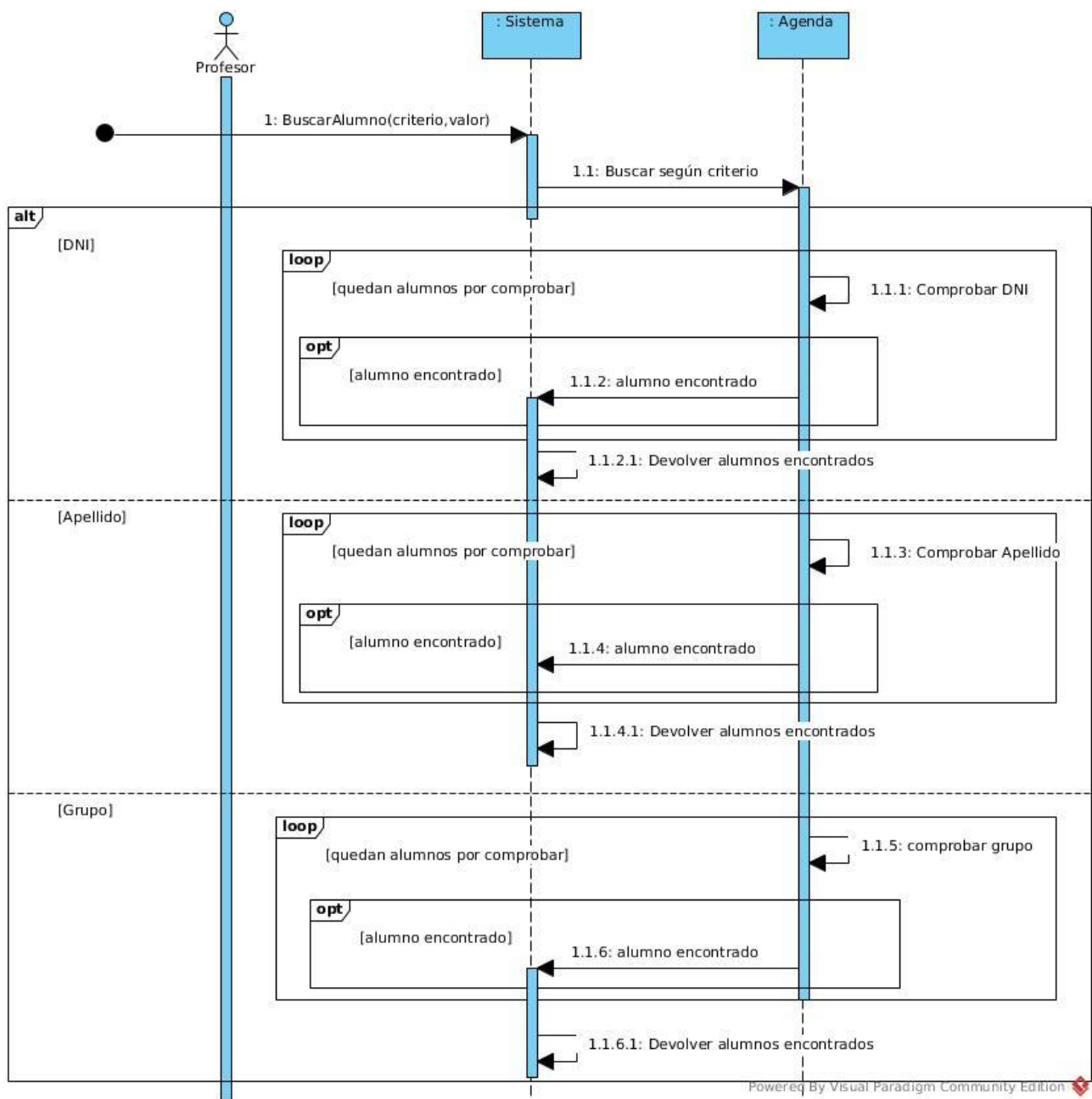


Figura 6.1 Diagrama de Secuencia CU-1

- Diagrama de secuencia CU-2.

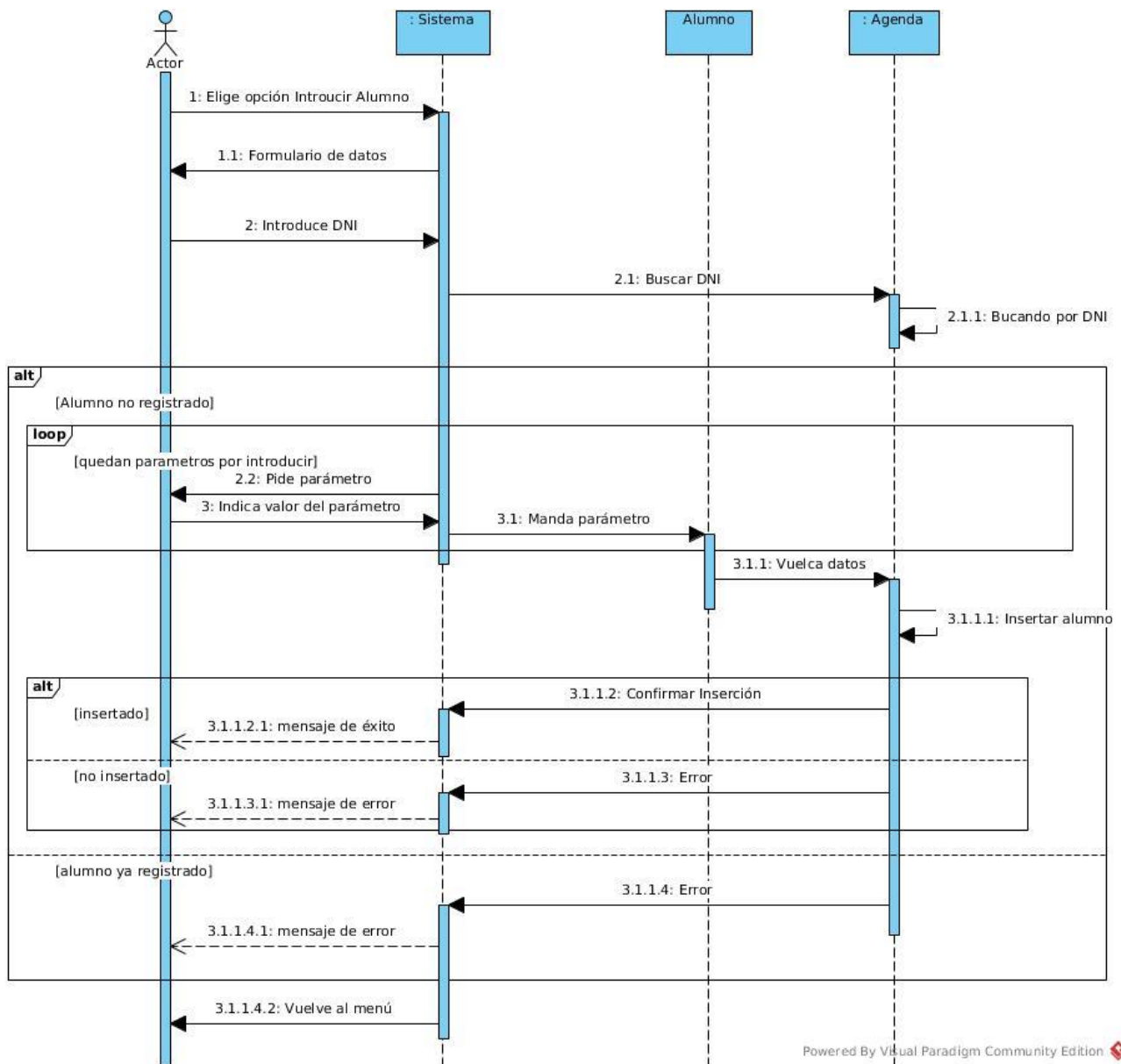


Figura 6.2 Diagrama de Secuencia CU-2

- **Diagrama de Secuencia CU-3**

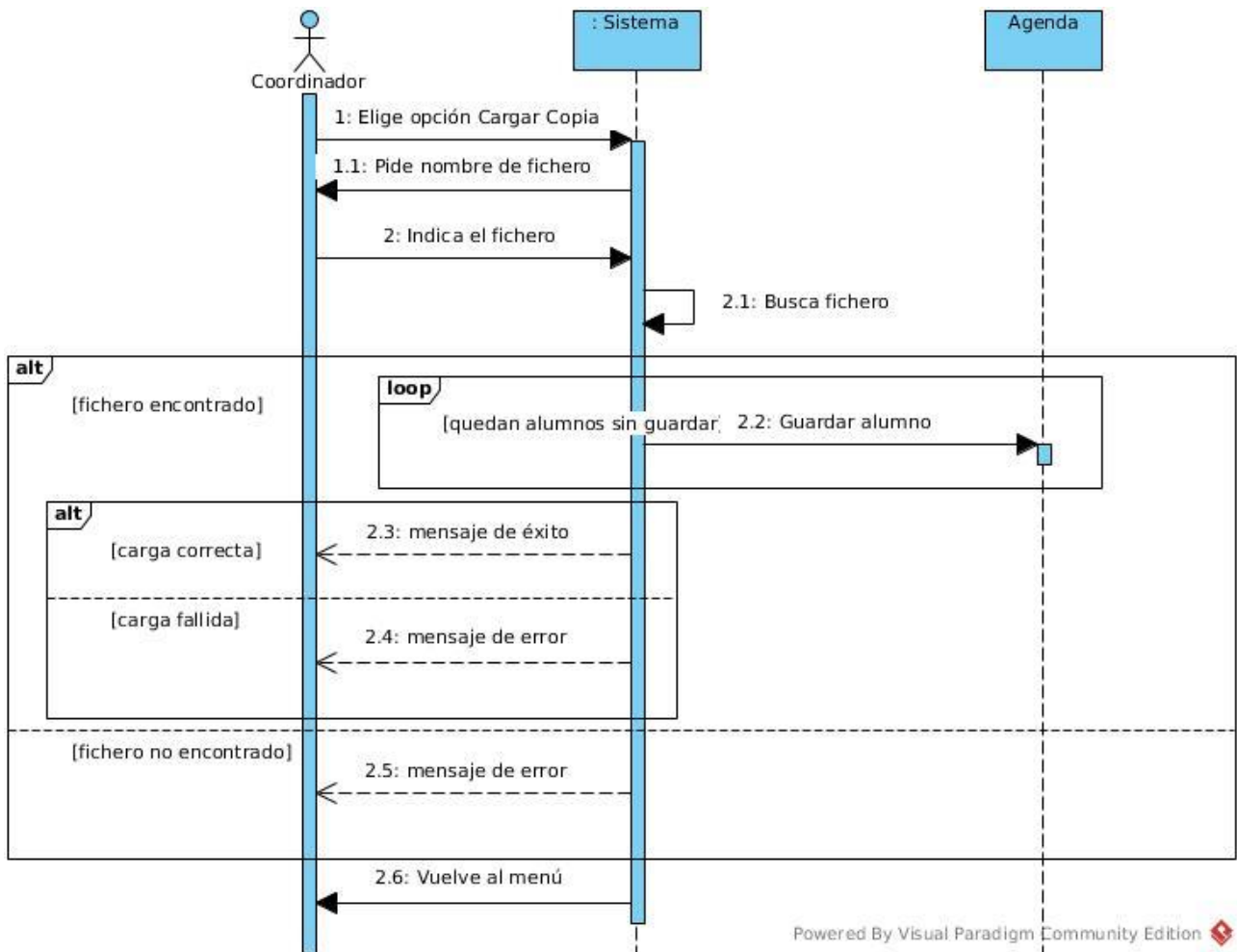


Figura 6.3 Diagrama de Secuencia CU-3

- Diagrama de Secuencia CU-4.

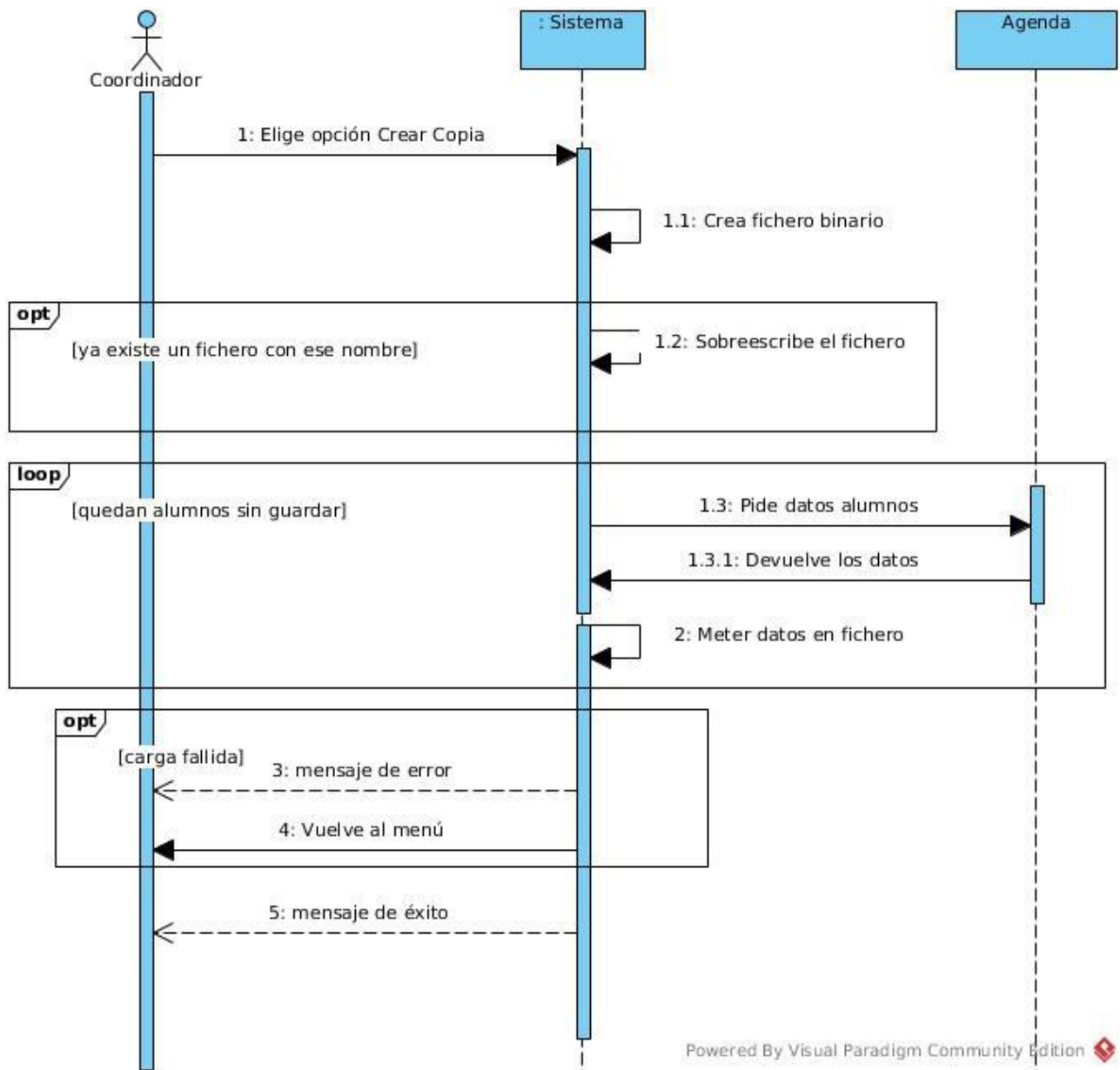


Figura 6.4 Diagrama de Secuencia CU-4

• Diagrama de Secuencia CU-5.

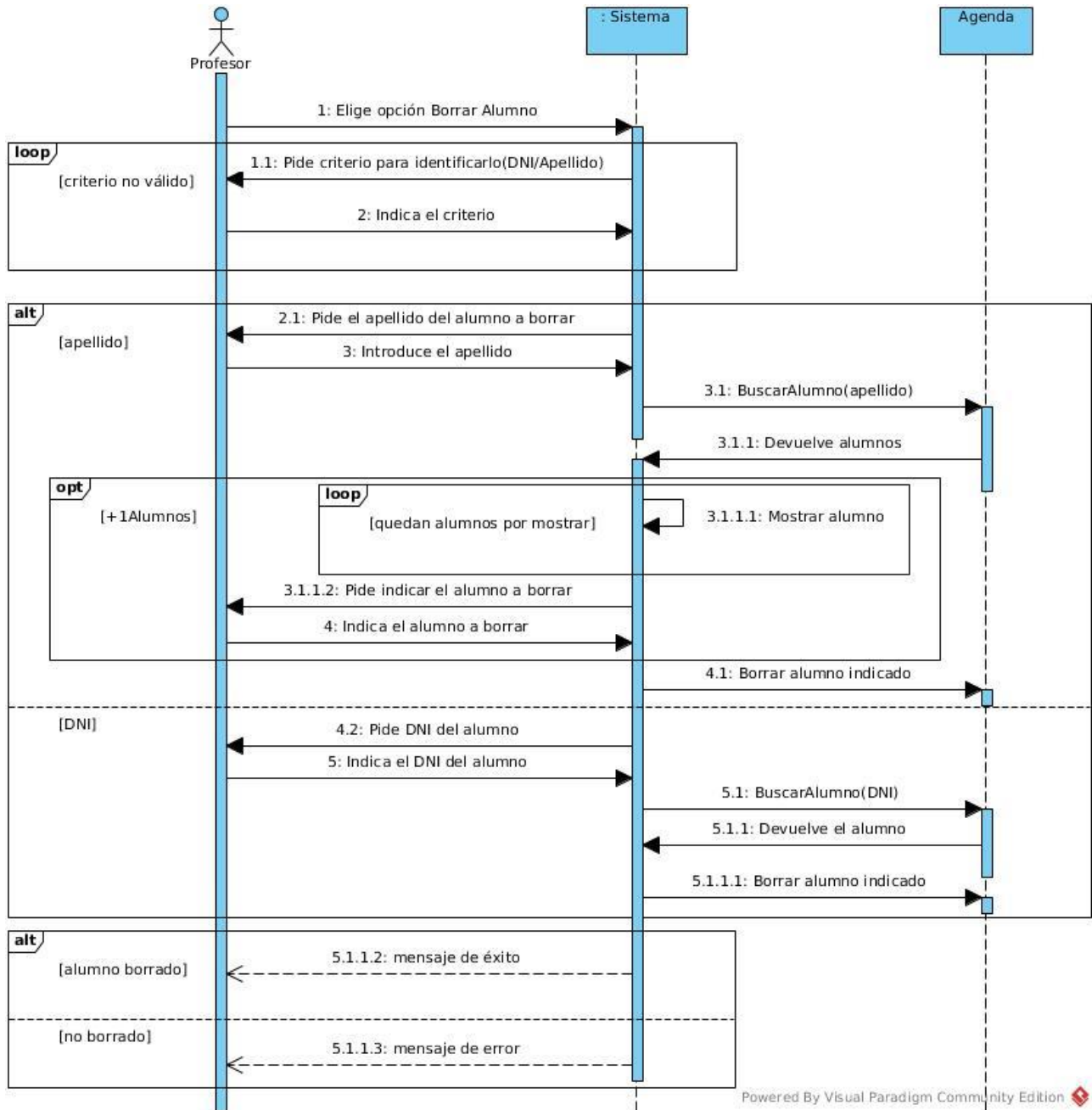


Figura 6.5 Diagrama de Secuencia CU-5

• Diagrama de Secuencia CU-6

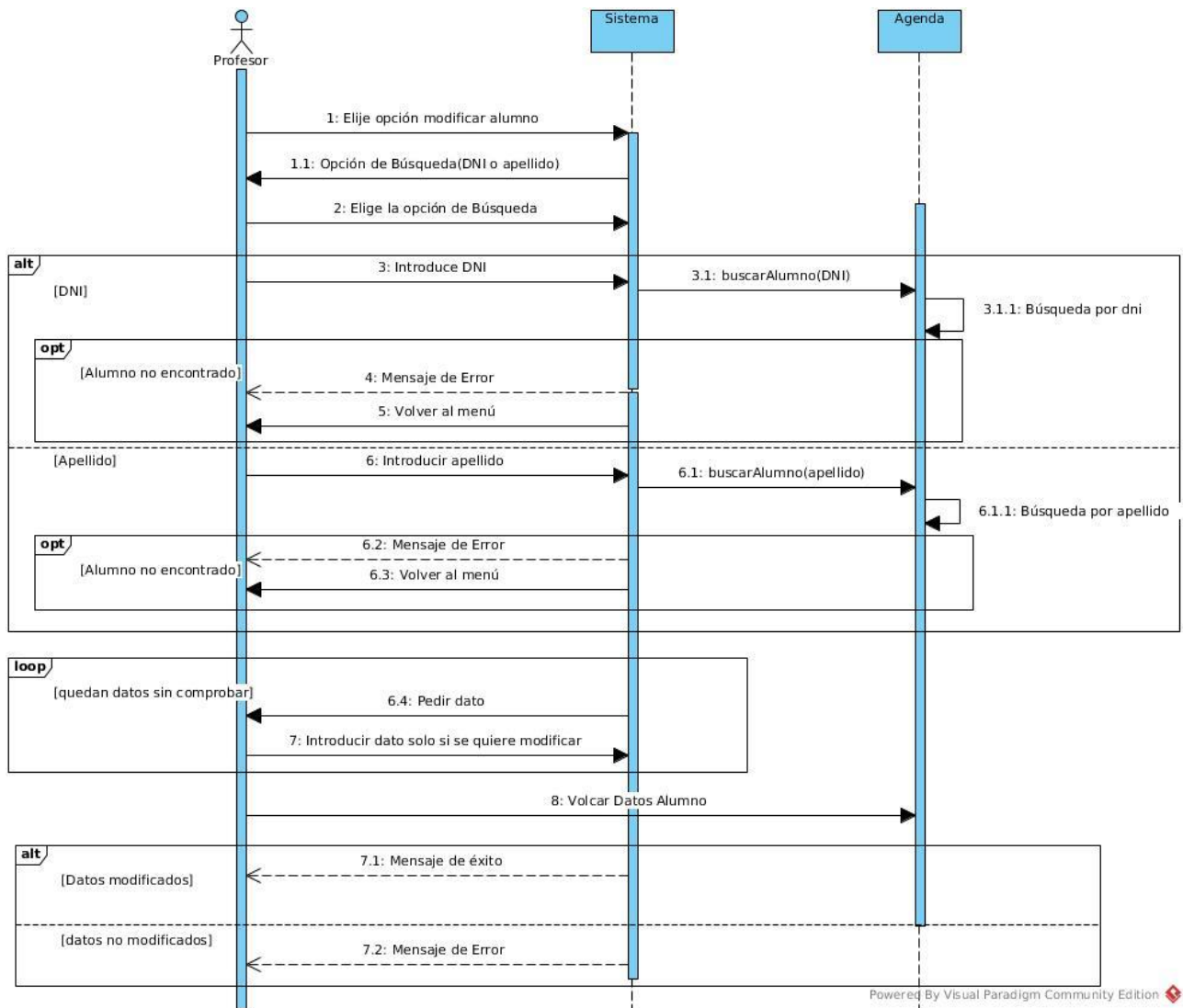


Figura 6.6 Diagrama de Secuencia CU-6

• Diagrama de Secuencia CU-7

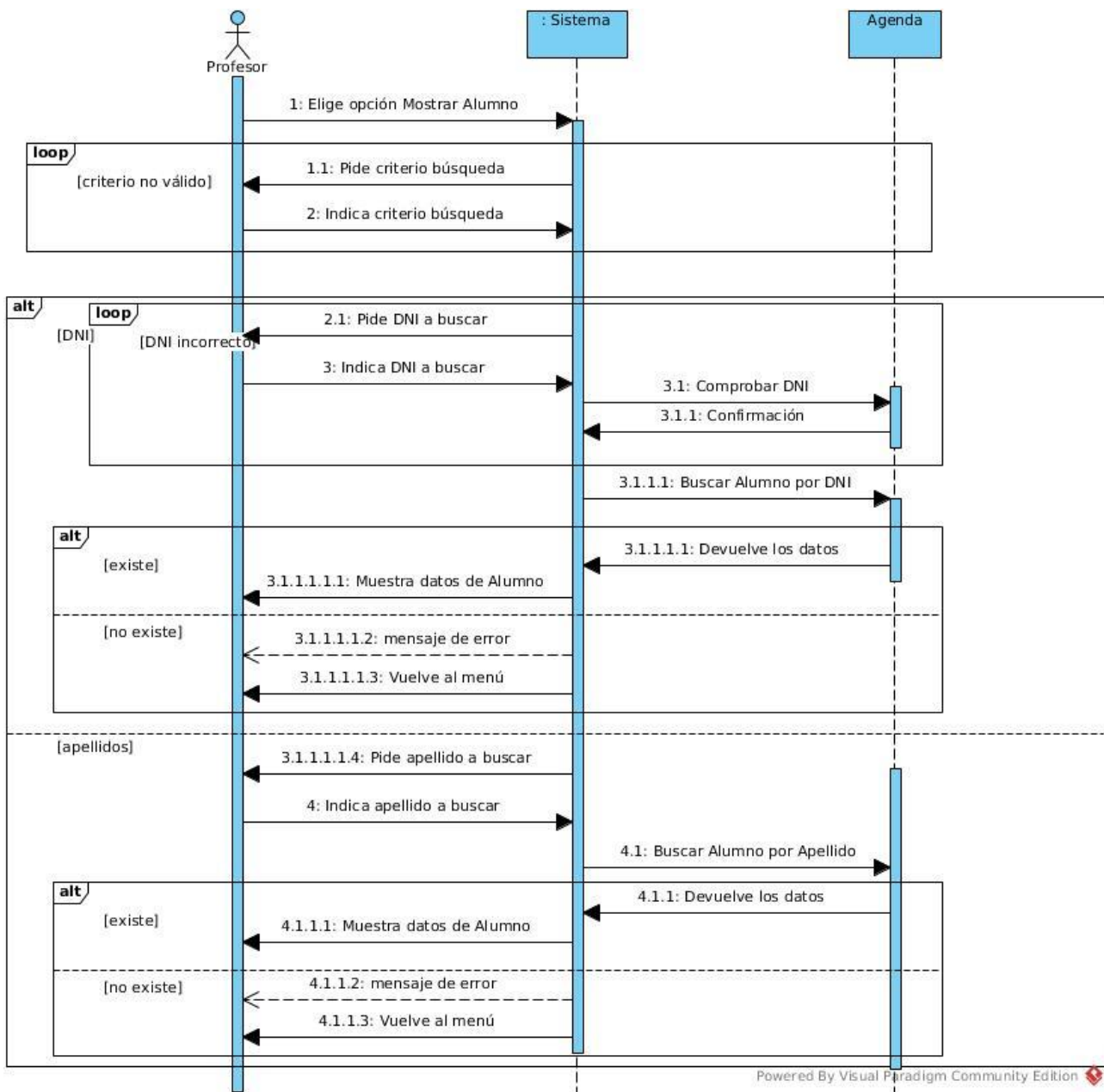


Figura 6.7 Diagrama de Secuencia CU-7

• Diagrama de Secuencia CU-8

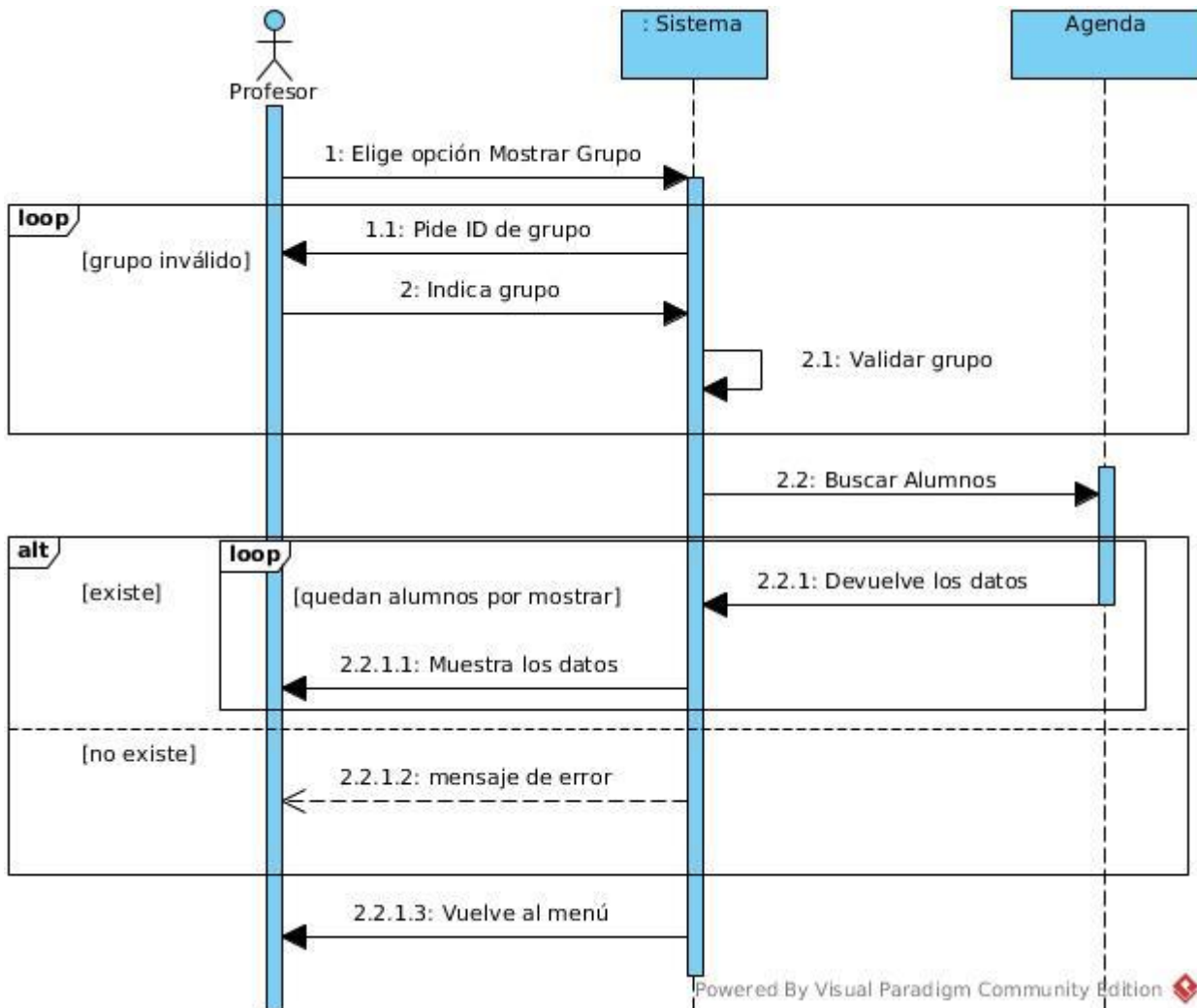


Figura 6.8 Diagrama de Secuencia CU-8

• Diagrama de Secuencia CU-9

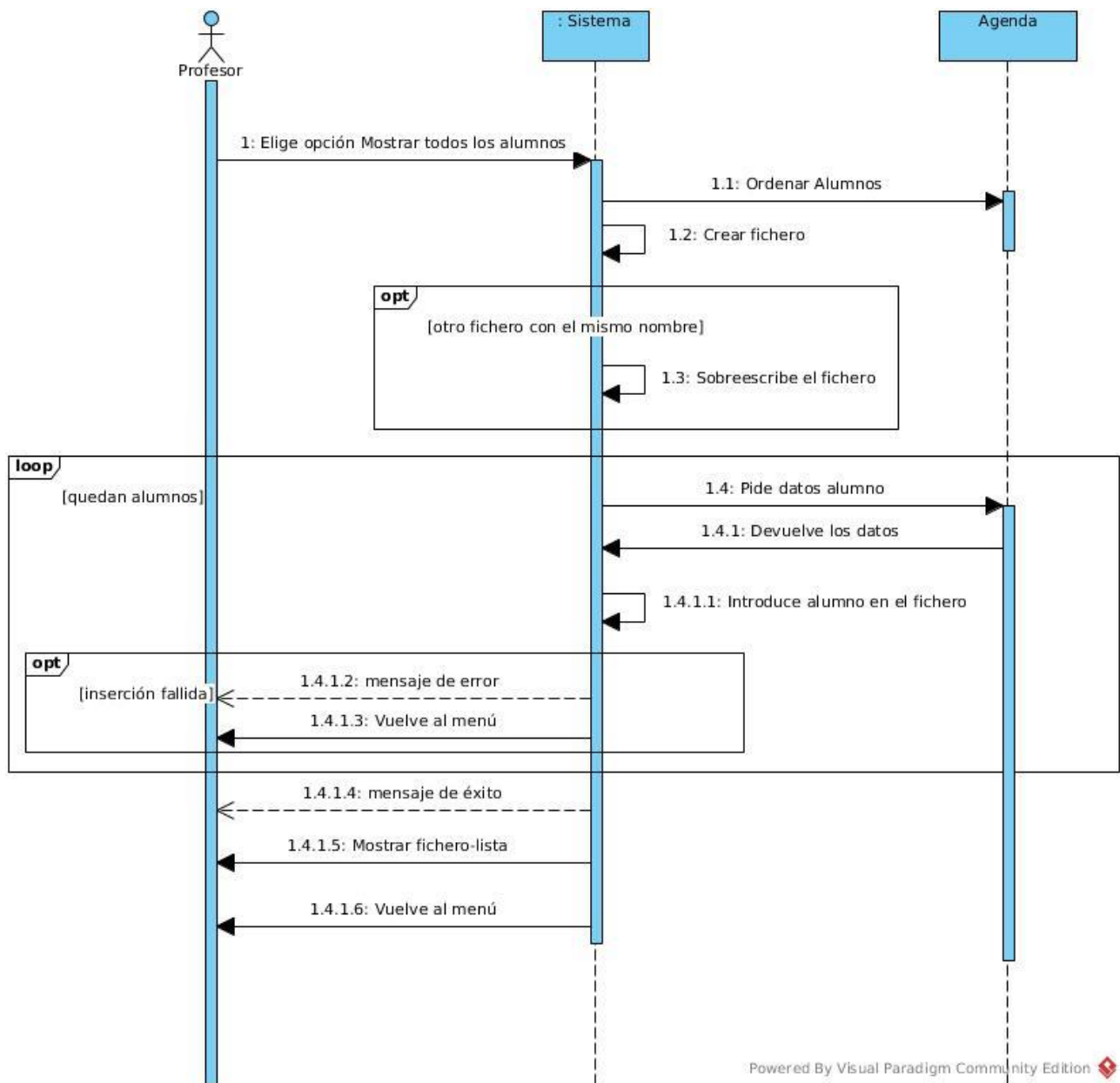


Figura 6.9 Diagrama de Secuencia CU-9

• Diagrama de Secuencia CU-10

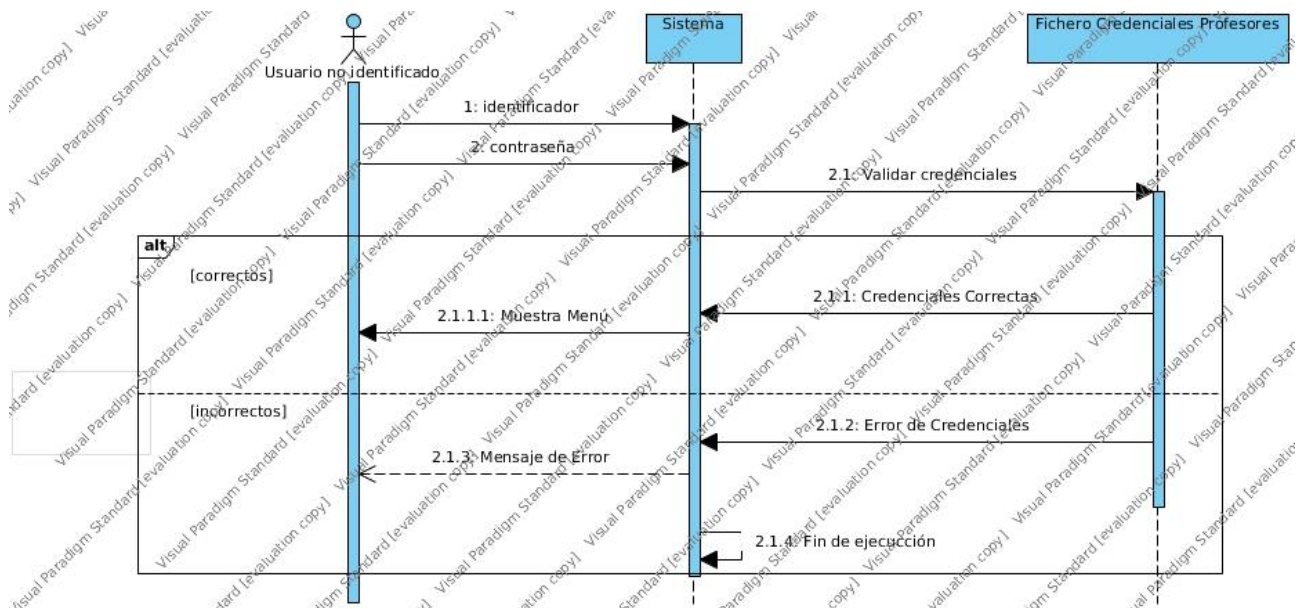


Figura 6.10 Diagrama de Secuencia CU-10

- Diagrama de Secuencia 11 CU-11

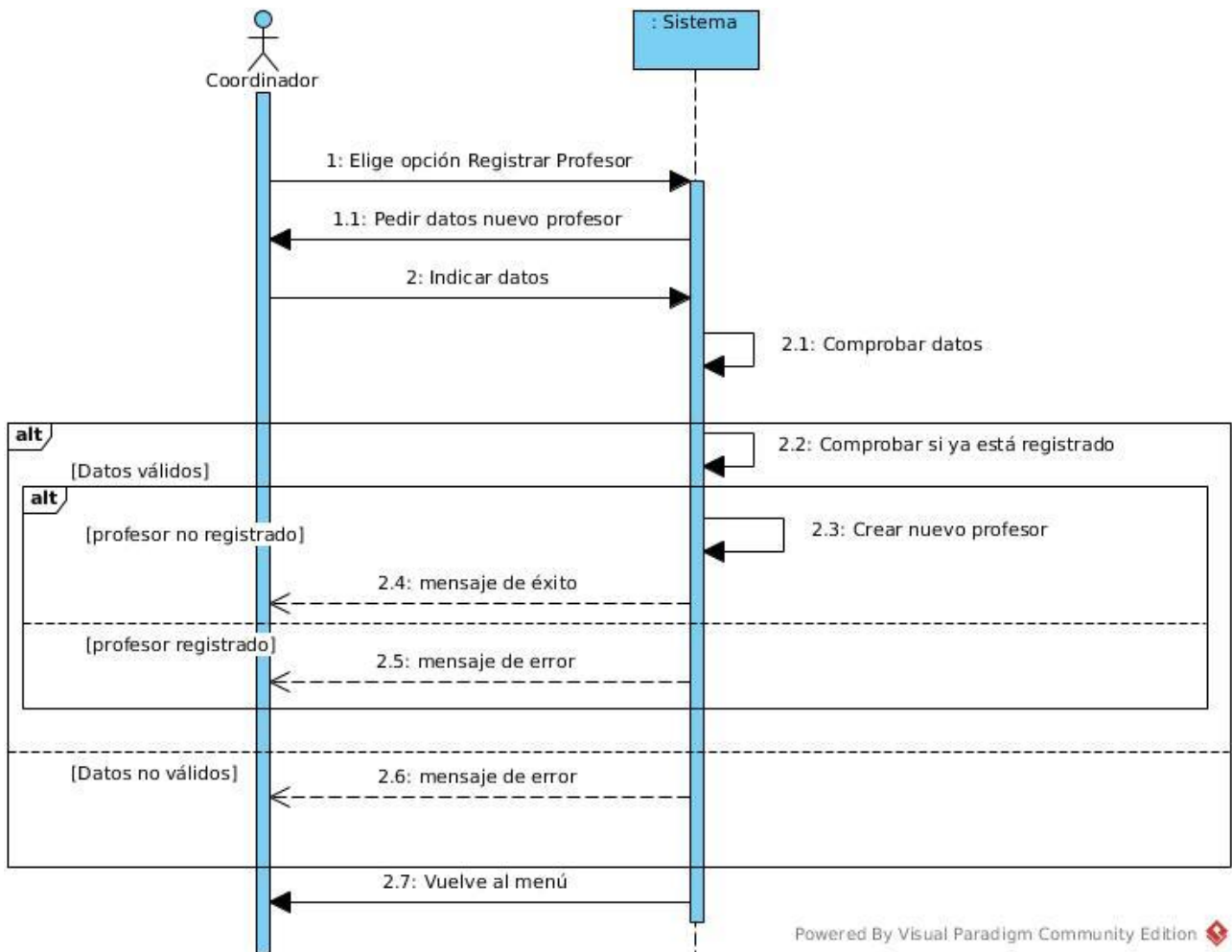


Figura 6.11 Diagrama de Secuencia CU-11

Capítulo 7. Diseño de la interfaz

7.1. Descripción de la interfaz

- **Menú de acceso o log in**

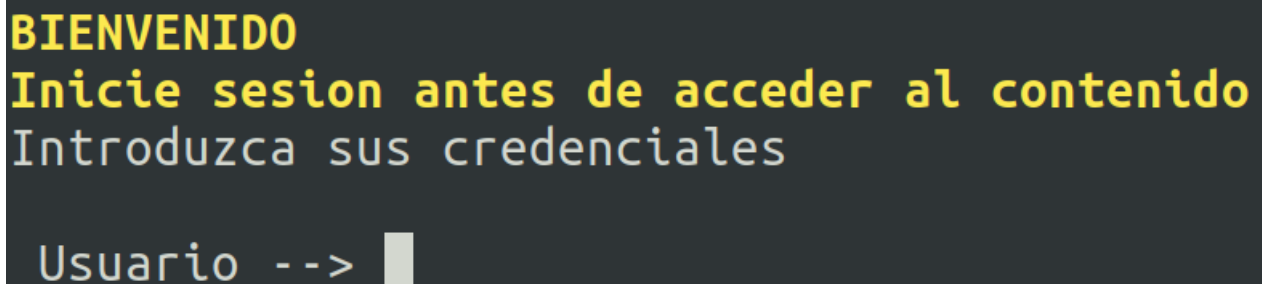
A dark gray rectangular area containing yellow and white text. The text reads: 'BIENVENIDO' in yellow, 'Inicie sesion antes de acceder al contenido' in yellow, 'Introduzca sus credenciales' in white, and 'Usuario --> ' followed by a white rectangular input field.

Figura 7.1 Prototipo de interfaz de menú de acceso

Éste es un prototipo básico del menú de acceso para iniciar sesión, cuando el usuario introduzca sus credenciales podrá acceder a nuestro menú principal previa validación de las mismas.

- Menú principal I Coordinador

Opciones del menú

[1] Cargar alumnos desde fichero
[2] Guardar alumnos en un fichero

[3] Introducir nuevo alumno
[4] Modificar datos de un alumno
[5] Eliminar un alumno

[6] Mostrar numero de alumnos registrados
[7] Mostrar datos de un alumno
[8] Mostrar alumnos de un grupo
[9] Mostrar lista de alumnos

[10] Registrar nuevo profesor

[11] Cargar copia de seguridad
[12] Crear copia de seguridad

[0] Volver

Opción:

Figura 7.2 Prototipo de interfaz de Menú Principal del coordinador

Éste es el prototipo de menú principal I donde nos encontramos con las distintas opciones a las que puede acceder cualquier usuario que esté registrado como coordinador de la asignatura.

- Menú principal II ayudante.

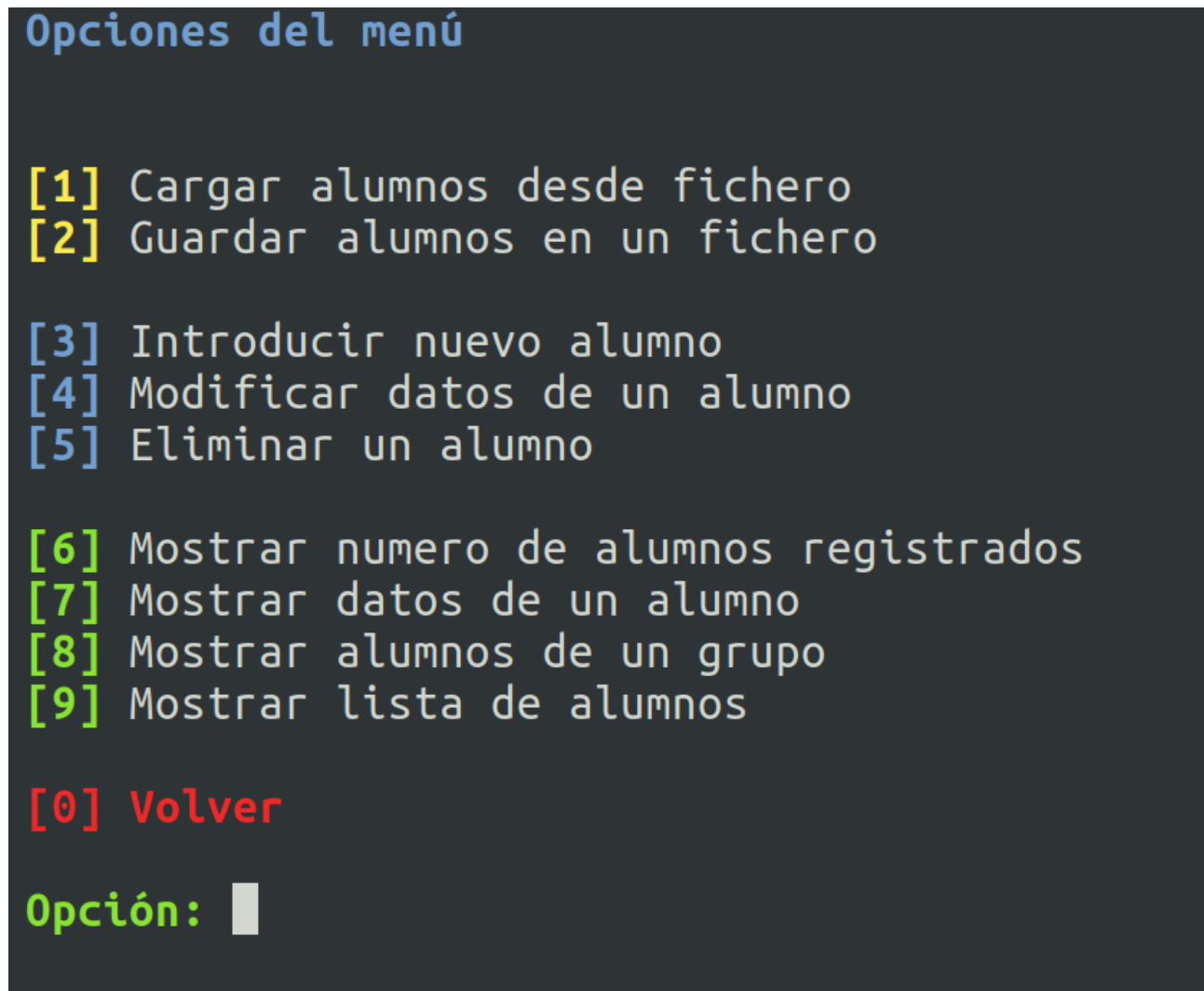


Figura 7.3 Prototipo de interfaz Menú Principal ayudante.

Éste es el prototipo de menú principal II donde nos encontramos con las distintas opciones a las que puede acceder el usuario que se encuentre registrado como ayudante.

A diferencia del menú principal I, observamos que el ayudante del coordinador no puede registrar a un nuevo profesor, realizar copias de seguridad cargar copias de seguridad.

Capítulo 8. Metodología SCRUM

Durante el desarrollo de este capítulo se expondrán los datos relacionados con la metodología SCRUM aplicada en el desarrollo de este sistema.

8.1 Product Backlog.

HU-1 Buscar Alumno

Prioridad: 0

Tiempo estimado: 6h

Funciones Auxiliares

Prioridad: 0

Tiempo estimado: 5h

HU-7 Mostrar Alumno

Prioridad: 1

Tiempo estimado: 4h

HU-2 Insertar Alumno

Prioridad: 1

Tiempo estimado: 6h

Informatización de Alumnos

HU-6 Modificar Alumno

Prioridad: 2

Tiempo estimado: 6h

HU-5 Borrar Alumno

Prioridad: 2

Tiempo estimado: 6h

HU-9 Mostrar Todos los Alumnos

Prioridad: 3

Tiempo estimado: 5h

HU-8 Mostrar Grupo

Prioridad: 4

Tiempo estimado: 5h

HU-3 Cargar Copia

Prioridad: 5

Tiempo estimado: 5h

HU-4 Crear Copia

Prioridad: 5

Tiempo estimado: 5h

HU-10 Log in

Prioridad: 6

Tiempo estimado: 7h

HU-11 Registrar Profesor

Prioridad: 6

Tiempo estimado: 7h

8.2 Sprint Backlogs

8.2.1 Sprint Backlog Semana 1

Sprint Backlog Semana 1

Primera reunión

5-12-2018

Francisco Vera Herencia - 3 horas

Rafael Piqueras Espinar - 2 horas

Segunda reunión

9-12-2018

Francisco Vera Herencia - 1 hora

Rafael Piqueras Espinar - 5 horas

Figura 8.1 Sprint Backlog Semana 1

8.2.2 Sprint Backlog Semana 2

Sprint Backlog Semana 2

Primera reunión

12-12-2018

Francisco Vera Herencia - 7 horas - Buscar Alumno, Funciones Auxiliares

Rafael Piqueras Espinar - 3 horas - Introducir alumno

Segunda reunión

14-12-2018

Francisco Vera Herencia - 13 horas - Buscar Alumno, Mostrar Alumno, Introducir alumno, Listar Todos los Alumnos

Rafael Piqueras Espinar - 8 horas - Borrar Alumno

Tercera reunión

16-12-2018

Francisco Vera Herencia - 20 horas - Mostrar Grupo, Cargar Copia, Crear Copia, Log in

Rafael Piqueras Espinar - 12 horas - Crear Copia, Registrar Profesor

Figura 8.2 Sprint Backlog Semana 2

8.3 Burndown Chart.

Hoja1

Función	Tiempo	Fecha reunión	Horas empleadas	Horas restantes
Buscar	6	28/11/18	0	67
Auxiliares	5	05/12/18	5	62
Mostrar	4	09/12/18	6	56
Insertar	6	12/12/18	10	46
Modificar	6	14/12/18	21	25
Borrar	6	16/12/18	32	0
MostrarTodos	5			0
MostrarGrupo	5			0
CargarCopia	5			0
CrearCopia	5			0
Login	7			0
Registro	7			0
Horas totales	67			

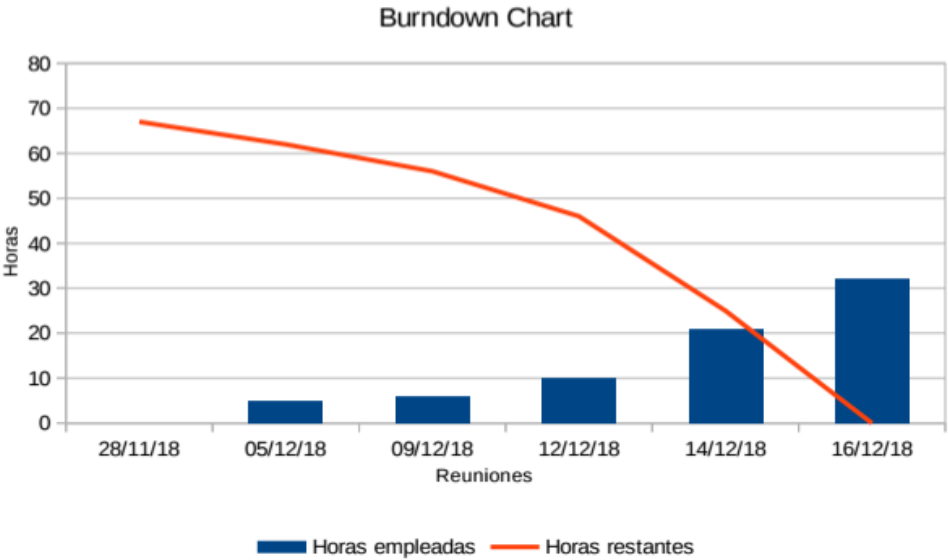


Figura 8.3 Burndown Chart

Capítulo 9. Técnicas de Validación

En este capítulo se van a validar, mediante matrices de validación, los diferentes aspectos desarrollados en el sistema, es decir se va a comprobar que el sistema resultante satisface los requisitos establecidos, tales como los requisitos funcionales en contraste con los casos de uso.

9.1 Matriz de requisitos funcionales/casos de uso.

Casos de uso Requisitos	CU-1	CU-2	CU-3	CU-4	CU-5	CU-6	CU-7	CU-8	CU-9	CU-10	CU-11
RF-1	X	X			X	X	X	X			
RF-2		X									
RF-3			X								
RF-4				X							
RF-5					X						
RF-6						X					
RF-7							X				
RF-8								X			
RF-9									X		
RF-10										X	
RF-11											X

Tabla 9.1 Matriz de Requisitos funcionales/casos de uso

9.2 Matriz de Casos de Uso/Clases.

Casos de Uso \ Clases	ListaAlumnos	Profesor	Alumno
CU-1	X		X
CU-2	X		X
CU-3	X	X	
CU-4	X	X	
CU-5	X		X
CU-6	X		
CU-7	X		
CU-8	X		
CU-9	X		
CU-10		X	
CU-11		X	

Tabla 9.2 Matriz de Casos de Uso/Clases

Apéndices

A. Bibliografía

Para la elaboración del presente documento, se han empleado las diferentes fuentes de consulta:

Lucidchart . Lucidchar, @2018Lucid sotwareInc [consulta: 5 octubre 2018]. Disponible en : <https://lucidchart.com/>

Markdowpreview. Markdownpreview @2018 [consulta: 10 octubre 2018]. Disponible en: <https://markdownlivepreview.com/>

Visual paradigm. Visual paradigm, ©2018 [consulta: 30 octubre 2018]. Disponible en: <https://www.visual-paradigm.com/>