



**Departamento de Informática**  
Universidad Técnica Federico Santa María



## Entregable II

### Análisis y Diseño de Software / Fundamentos de Ingeniería de Software

Integrantes:

Nombre	Email	Teléfono
Francisco Alvial	francisco.alvial.13@sansano.usm.cl	+56961595115
Alejandra López	alejandre.lopez.13@sansano.usm.cl	+56978782026
Marcom Moyano	marcom.moyano.13@sansano.usm.cl	+56957413169

## INTRODUCCIÓN

Como se ha visto durante las clases, los requerimientos son unas de las cosas más importantes a tener en consideración durante el desarrollo de la solución del problema planteado por nuestro cliente. Es por esto que siempre es mejor hacer la toma de requerimientos en grupo y no sólo una persona, para poder entender bien el problema del cliente y así encontrar la mejor solución.

Además, como se trabajará en grupo se debe ser muy claro a la hora de concluir y desarrollar las cosas, para que no haya ambigüedad y puede ser entendible por todos.

En el presente documento se encuentra lo que nuestro grupo concluyó en base a lo solicitado por el cliente, primero se encuentran los requerimientos solicitados, luego una serie de casos de uso con una breve descripción, luego el detalle de 4 de los casos de uso anteriores, luego el diagrama de todos los casos de uso, luego 2 diagramas de secuencia de 2 casos de uso de los 4 detallados y finalmente el modelo relacional correspondiente a la base de datos.

## REQUISITOS

Se agregó en los requisitos la funcionalidad principal del proyecto, la cual es que es una página web que muestra contenidos para el aprendizaje. También se juntaron algunos de los requisitos, los cuales estaban relacionados entre sí.

Funcionales:

- Plataforma que muestra contenido siendo capaz de administrarlo, la cual debe ser adaptable según el tipo de aprendizaje.
- El usuario se registra ingresando sus datos y respondiendo la encuesta (una única vez) para determinar el tipo de persona. Luego de registrarse es capaz de administrar su perfil, el cual tiene un encabezado característico para cada uno de estos.
- La plataforma debe entregar la información según el tipo de persona. Luego de revisar su perfil puede revisar los otros 3 en un orden determinado.

No Funcionales:

- Administrador puede agregar módulos
- Utilizar Node.js, Angular.js, Express.js, MySQL, Github, Trello y Visual Paradigm.
- Sólo puede registrarse si posee un mail de la universidad.
- Sistema no muestra tipo de persona al alumno.

ID	REQUISITO	OBLIGATORIEDAD
FR1	Plataforma que muestra contenido, siendo capaz de administrarlo, la cual debe ser adaptable según el tipo de aprendizaje.	Obligatorio
FR2	El usuario se registra ingresando sus datos y respondiendo la encuesta (una única vez) para determinar el tipo de persona. Luego de registrarse es capaz de administrar su perfil, el cual tiene un encabezado característico para cada uno de estos.	Obligatorio

## Entregable 2

FR3	La plataforma debe entregar la información según el tipo de persona. Luego de revisar su perfil puede revisar los otros 3 en un orden determinado.	Obligatorio
-----	--	-------------

ID	REQUISITO NF	ASOCIADO A
NFR1	Utilizar Node.js, Angular.js, Express.js, MySQL, Github, Trello y Visual Paradigm.	FR1
NFR2	Administrador puede agregar módulos	FR1, FR3
NFR3	Sólo puede registrarse si posee un mail de la universidad	FR2
NFR4	Sistema no muestra tipo de persona al alumno.	FR2

## **CASOS DE USO**

La siguiente numeración representa los casos de uso extraídos de los requerimientos del proyecto, de los cuales se escogieron los 4 más complejos para detallarlos:

Caso 1, Registrar cuenta: **Alumno** ingresa sus datos para pertenecer al **sistema**, y este los verifica e ingresa a la base de datos.

Caso 2, Editar perfil: **Alumno** o **profesor** editan información de su propio perfil y el **sistema** los actualiza en la base de datos.

Caso 3, Responder Encuesta: **Alumno** responde la encuesta al momento de registrarse y el **sistema** guarda el tipo de persona de acuerdo a sus respuestas a la base de datos.

Caso 4, Seleccionar Tema: **Alumno** o **profesor** selecciona un tema el cual revisará y el **sistema** lo muestra por pantalla.

Caso 5, Administrar Materia: **Administrador** o **profesor** agregan, editan o eliminan alguna materia y el **sistema** la procesa de acuerdo a la selección en la base de datos.

Caso 6, Descargar Materia: **Administrador**, **profesor** o **alumno** descargan el material que contiene la página, donde el **sistema** convierte el material en un archivo pdf y lo descarga.

Caso 7, Administrar cuenta: **Administrador** crea, edita, elimina o revisa la cuenta de algún usuario y el **sistema** la procesa de acuerdo a la selección en la base de datos.

Caso 8, Cambiar Tipo: **Alumno** cambia forma de ver material una vez que haya revisado la forma en que se muestra según su tipo, donde el **sistema** muestra en pantalla el orden de la materia de forma diferente al tipo actual.

Caso 9, Recuperar Contraseña: **Alumno** recupera su contraseña haciendo click en recuperar contraseña y el **sistema** manda un mail con una contraseña aleatoria, la cual puede cambiarla cuando desee.

Los 4 casos de uso escogidos fueron: caso 1, caso 3, caso 5 y caso 7.

## CASOS DE USO COMPLETO

Caso de uso: Registrar cuenta	
Propósito: Realizar un registro en el sistema con los datos del alumno	
Precondiciones: No hay	
Postcondiciones: Sistema redirige a la página de inicio.	
Actores: Alumno	
DESCRIPCIÓN DETALLADA CASO EXITOSO	
Alumno	Sistema
1. Este caso de uso comienza cuando el alumno quiere registrarse dentro de la página.	
2. El alumno selecciona registrarse.	
	3. El sistema muestra una serie de campos los cuales deben ser llenados con la información del usuario.
4. El alumno llena correctamente todos los campos solicitados.	
5. El alumno selecciona "Registrar".	
	6. El sistema verifica la correctitud de los datos ingresados.
	7. *
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS	
7.a Alumno no responde correctamente todos los campos solicitados o deja alguno en blanco, el proceso vuelve a 2.	
7.b. El alumno ya se encuentra registrado.	
	7.b.1. El sistema informa que el usuario ya existe, el proceso vuelve a 2.

\* = En esta parte comienza el caso de uso Responder encuesta, el cual se detalla más abajo.

Administrar cuenta
Caso de uso: Crear cuenta
Propósito: Crear una cuenta nueva (profesor)
Precondiciones: El administrador debe estar logueado y no debe existir con anterioridad el usuario a crear.
Postcondiciones: Sistema redirecciona al perfil de la cuenta creada.
Actores: Administrador
DESCRIPCIÓN DETALLADA CASO EXITOSO

## Entregable 2

Administrador	Sistema
1. Este caso de uso comienza cuando el administrador quiere crear una nueva cuenta.	
2. El administrador selecciona crear cuenta.	
	3. El sistema muestra formulario con los campos a llenar.
4. El usuario llena los campos solicitados.	
5. El usuario pulsa crear.	
	6. El sistema verifica la correctitud de los datos ingresados.
	7. El sistema muestra un mensaje de éxito y termina el proceso
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS	
4.a. El usuario que intenta crear ya existe.	
	4.a.1.El sistema informa que el usuario ya existe, el proceso vuelve a 2.
4.b. El usuario olvidó completar un campo o no ingresó correctamente la información.	
	4.b.1. El sistema informa el error, el proceso vuelve a 2.
<b>Caso de uso: Eliminar cuenta</b>	
Propósito: Eliminar cuenta de un usuario.	
Precondiciones: El administrador debe estar logueado y la cuenta a eliminar debe existir.	
Postcondiciones:	
Actores: Administrador	
DESCRIPCIÓN DETALLADA CASO EXITOSO	
Administrador	Sistema
1. Este caso de uso comienza cuando el administrador quiere eliminar una cuenta.	
	2. Sistema muestra todos los usuarios registrados.
3. El administrador pulsa eliminar en el usuario que quiere quitar.	
	4. El sistema solicita una confirmación de lo requerido.
5. El administrador responde la solicitud.	
6. El administrador confirma la solicitud.	
	7. El sistema elimina al usuario, el proceso termina.
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS	
6.1. El administrador no confirma la solicitud.	
	6.1.1. El sistema no elimina al usuario, el proceso vuelve a 2.

## Entregable 2

### Caso de uso: Editar cuenta

Propósito: Editar datos de una cuenta existente.

Precondiciones: El administrador debe estar logueado y la cuenta a editar debe existir.

Postcondiciones: Sistema redirecciona al perfil de la cuenta editada.

Actores: Administrador

#### DESCRIPCIÓN DETALLADA CASO EXITOSO

Administrador	Sistema
1. Este caso de uso comienza cuando el administrador quiere editar una cuenta.	
	2. Sistema muestra todos los usuarios registrados.
3. El administrador pulsa editar en el usuario que quiere modificar.	
	4. El sistema muestra un formulario con los datos del usuario.
5. El administrador cambia los datos que desee.	
6. El administrador presiona guardar cambios, el proceso termina	
	7. El sistema muestra mensaje de éxito, el proceso termina.

#### DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS: No hay

### Caso de uso: Ver perfil

Propósito: Ver perfil con los datos de un usuario

Precondiciones: El administrador debe estar logueado y la cuenta a ver debe existir.

Postcondiciones: Sistema redirecciona al perfil de la cuenta seleccionada.

Actores: Administrador

#### DESCRIPCIÓN DETALLADA CASO EXITOSO

Administrador	Sistema
1. Este caso de uso comienza cuando el administrador quiere ver una cuenta.	
	2. Sistema muestra todos los usuarios registrados.
3. El administrador pulsa ver en el usuario que quiere mirar.	
	4. El sistema muestra el perfil del usuario seleccionado, el proceso termina.

#### DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS: No hay

### Caso de uso: Responder encuesta



## Entregable 2

Propósito: Responder encuesta de registro para saber el tipo de usuario que es.	
Precondiciones: El alumno debe haber ingresado sus datos para el registro	
Postcondiciones: Sistema redirecciona al perfil del alumno.	
Actores: Alumno	
DESCRIPCIÓN DETALLADA CASO EXITOSO	
Alumno	Sistema
1. Este caso de uso comienza cuando el alumno se está registrando, ingresó sus datos y seleccionó registrar.	
	2. El sistema muestra una serie de preguntas.
3. El alumno responde las preguntas.	
4. El alumno presiona terminar encuesta.	
	5. El sistema verifica la correctitud de sus respuestas.
	6. El sistema responde un mensaje de éxito, el proceso termina
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS	
3.1. El alumno no respondió alguna pregunta	
3.1.2. El alumno presiona terminar encuesta.	
	3.1.3. El sistema verifica la correctitud de sus respuestas.
	3.1.4. El sistema entrega mensaje de error, el proceso vuelve a 2.

Administrar materia	
Caso de uso: Agregar materia.	
Propósito: Agregar materia nueva.	
Precondiciones: El administrador o profesor que agregará materia debe estar logueado.	
Postcondiciones: Sistema redirecciona a vista de las materias existentes.	
Actores: Administrador, profesor	
DESCRIPCIÓN DETALLADA CASO EXITOSO	
Administrador, profesor	Sistema
1. Este caso de uso comienza cuando el administrador/profesor quiere agregar una materia.	
2. El administrador/profesor selecciona agregar materia.	
	3. El sistema muestra formulario con los campos a llenar.
4. El administrador/profesor llena los campos solicitados.	

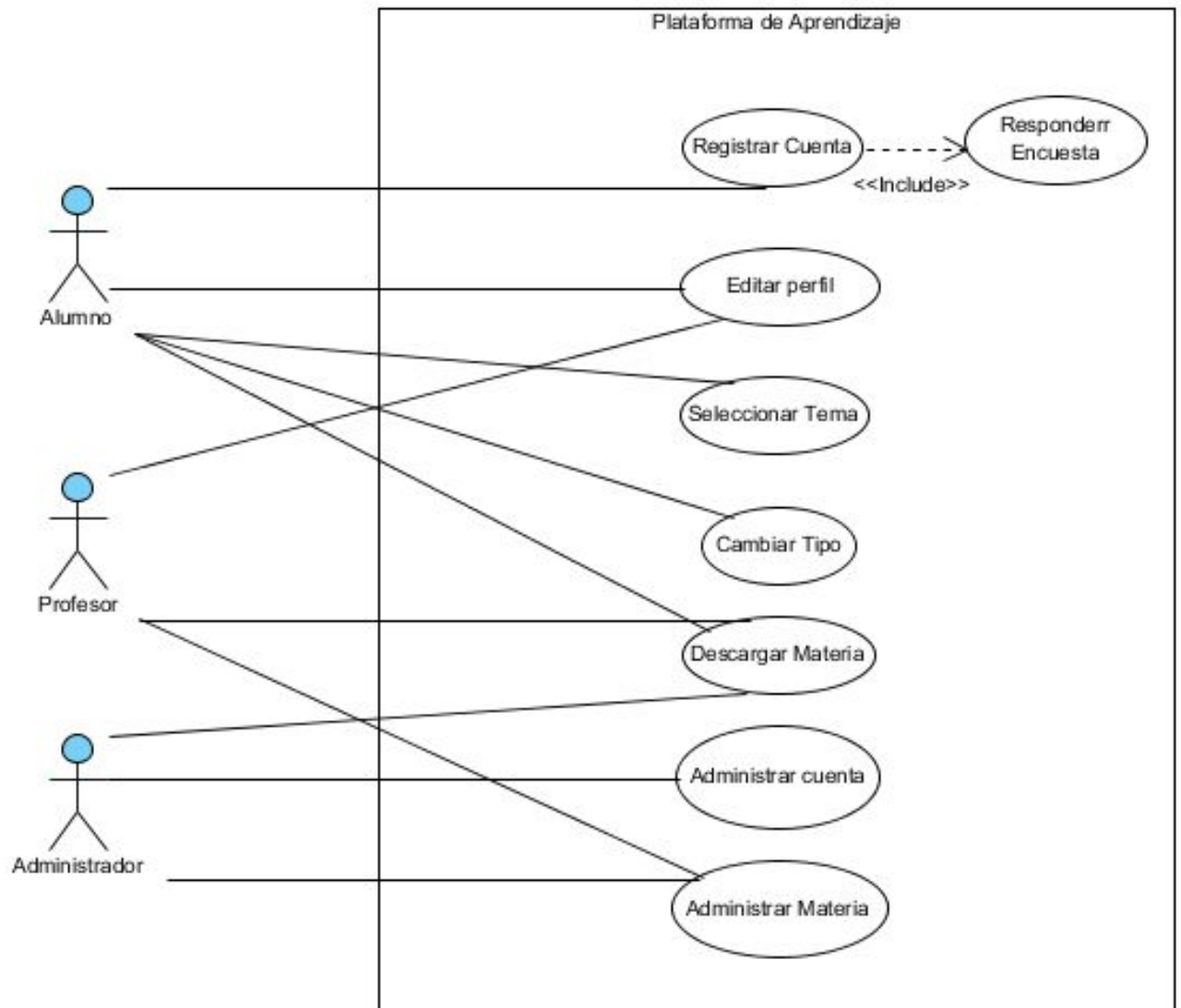
## Entregable 2

5. El administrador/profesor pulsa agregar.	
	6. El sistema verifica la correctitud de los datos ingresados.
	7. El sistema muestra un mensaje de éxito y termina el proceso
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS	
4.a. El administrador/profesor olvidó completar un campo o no ingresó correctamente la información.	
	4.a.1. El sistema informa el error, el proceso vuelve a 2.
<b>Caso de uso: Eliminar materia</b>	
Propósito: Eliminar materia de un ramo.	
Precondiciones: El administrador o profesor que eliminará materia debe estar logueado y la materia a eliminar debe existir.	
Postcondiciones: Sistema redirecciona a vista de las materias existentes.	
Actores: Administrador, profesor	
DESCRIPCIÓN DETALLADA CASO EXITOSO	
<b>Administrador, profesor</b>	<b>Sistema</b>
1. Este caso de uso comienza cuando el administrador/profesor quiere eliminar una materia.	
	2. Sistema muestra todas las materias existentes en el sistema.
3. El administrador/profesor pulsa eliminar en la materia que quiere quitar.	
	4. El sistema solicita una confirmación de lo requerido.
5. El administrador/profesor responde la solicitud.	
6. El administrador/profesor confirma la solicitud.	
	7. El sistema elimina la materia, el proceso termina.
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS	
6.1. El administrador/profesor no confirma la solicitud.	
	6.1.1. El sistema no elimina la materia, el proceso vuelve a 2.
<b>Caso de uso: Editar materia.</b>	

## Entregable 2

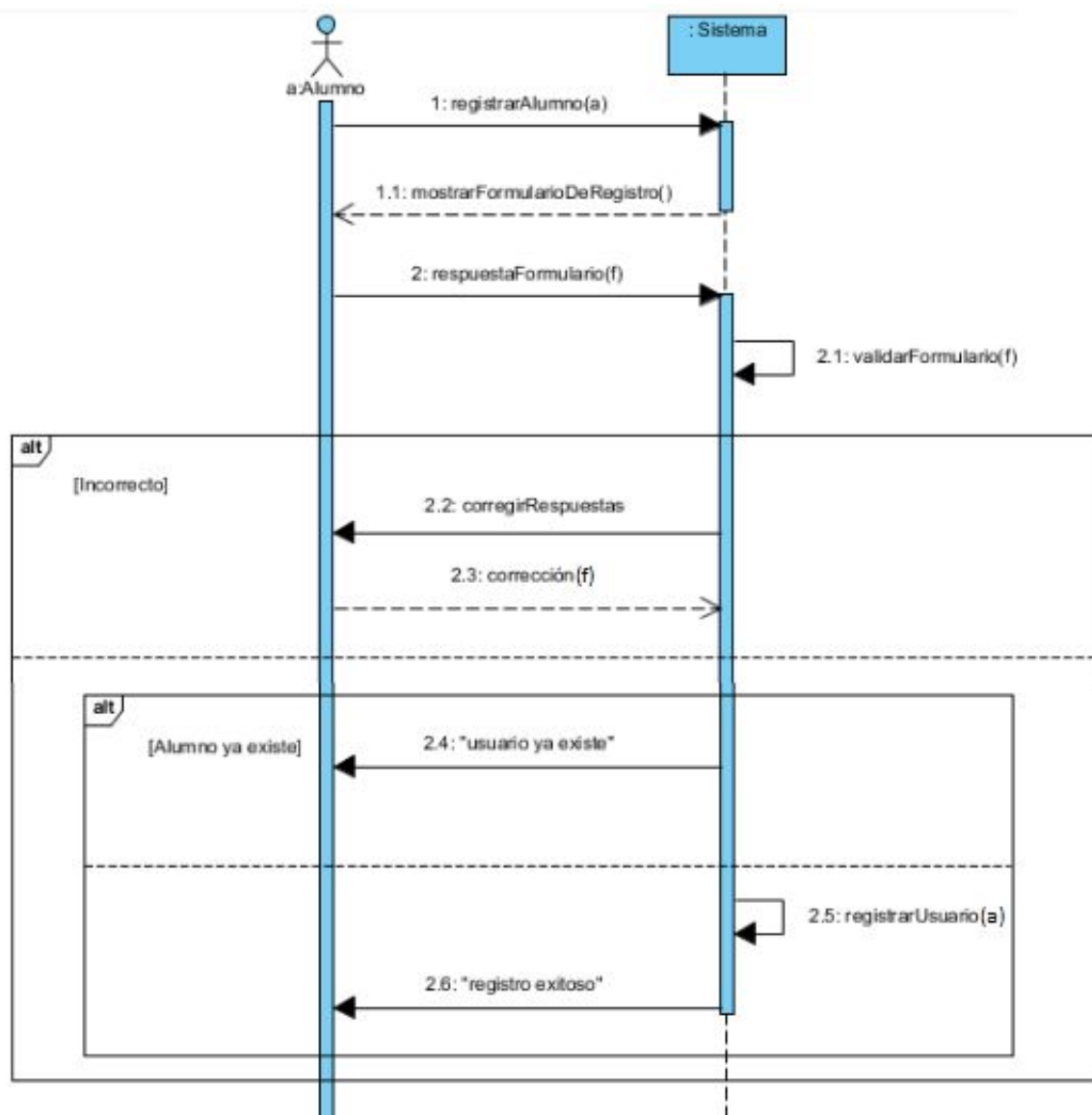
Propósito: Editar una materia existente.	
Precondiciones: El administrador/profesor debe estar logueado y la materia a editar debe existir.	
Postcondiciones: Sistema redirecciona a vista de las materias existentes.	
Actores: Administrador, profesor	
DESCRIPCIÓN DETALLADA CASO EXITOSO	
Administrador	Sistema
1. Este caso de uso comienza cuando el administrador/profesor quiere editar una materia.	
	2. Sistema muestra toda la materia registrada en el sistema.
3. El administrador pulsa editar en la materia que quiere modificar.	
	4. El sistema muestra la materia seleccionada.
5. El administrador/profesor cambia los datos que desee.	
6. El administrador/profesor presiona guardar cambios, el proceso termina	
	7. El sistema muestra mensaje de éxito, el proceso termina.
DESCRIPCIÓN DETALLADA CASOS ALTERNATIVOS: No hay	

# DIAGRAMA DE CASOS DE USO



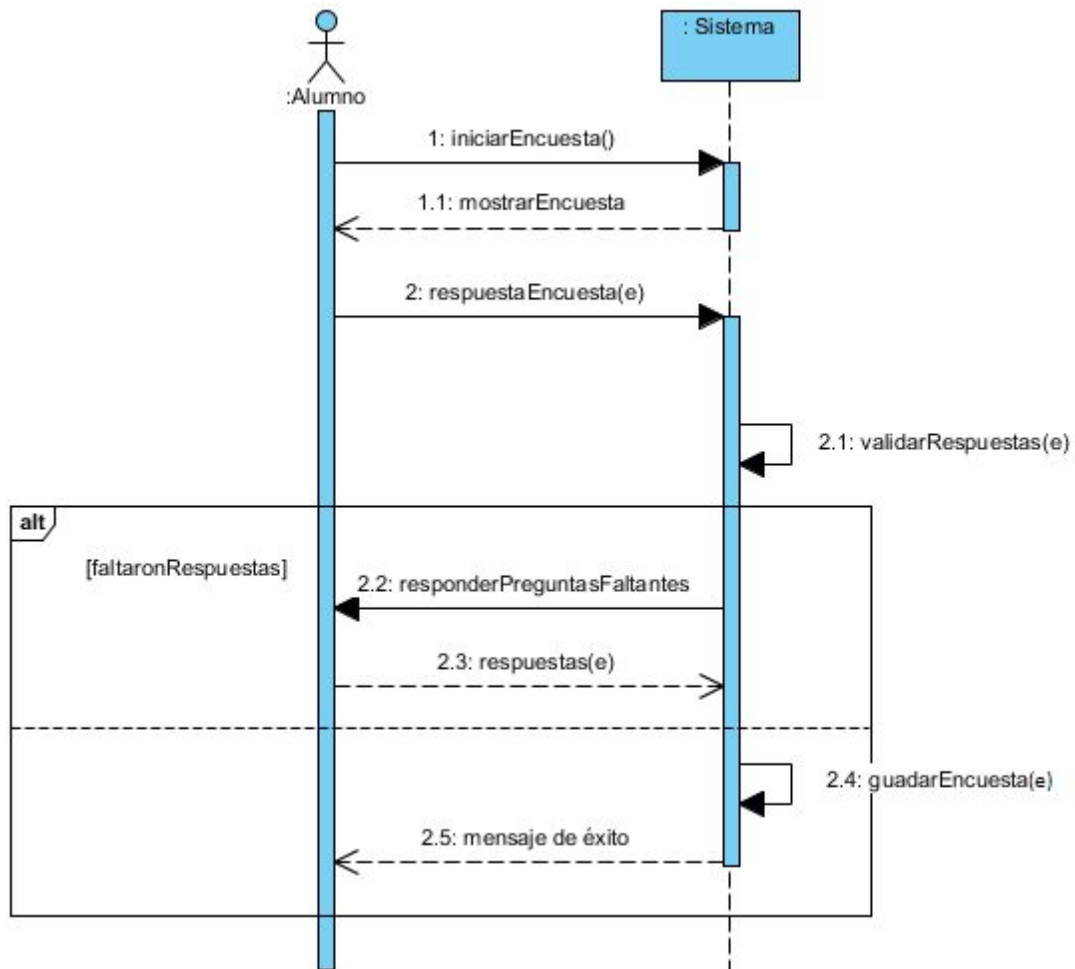
# DIAGRAMAS DE SECUENCIA DEL SISTEMA

Diagrama de secuencia del caso de uso Registrar cuenta:



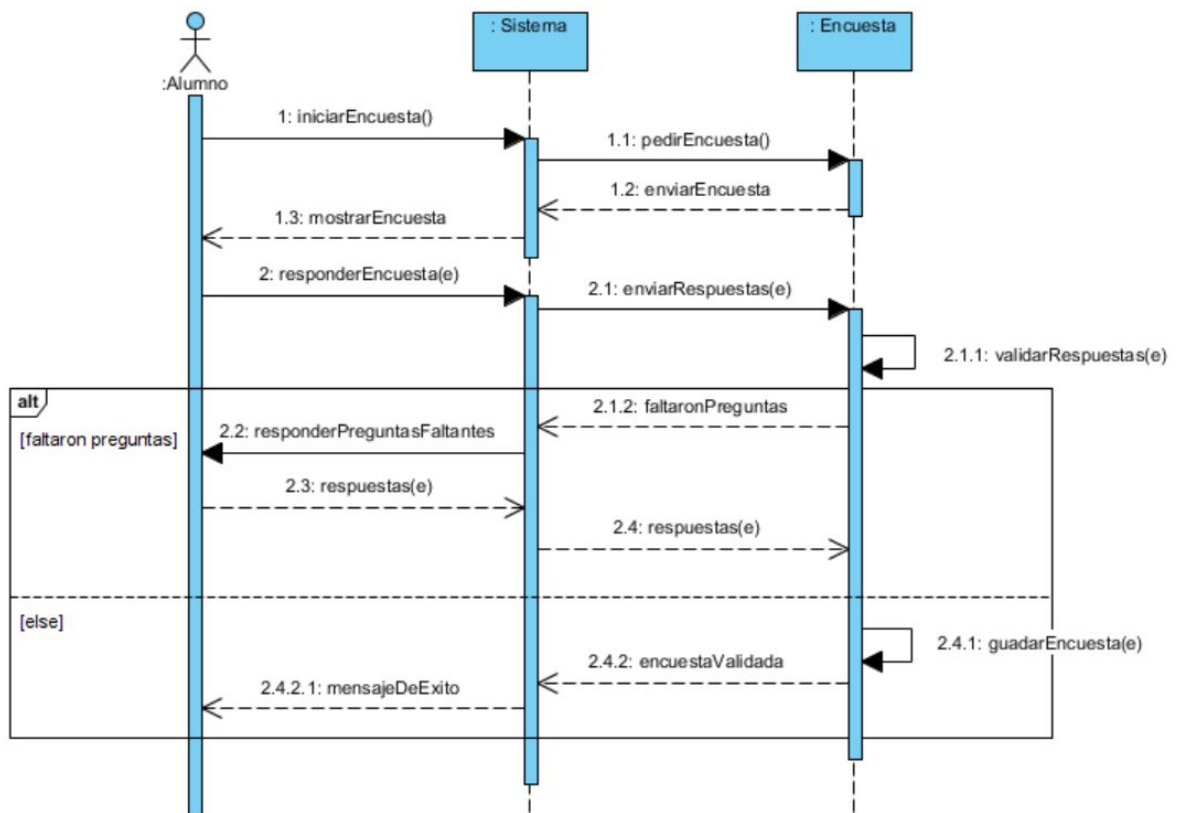
## Entregable 2

Diagrama de secuencia del caso de uso Responder encuesta:



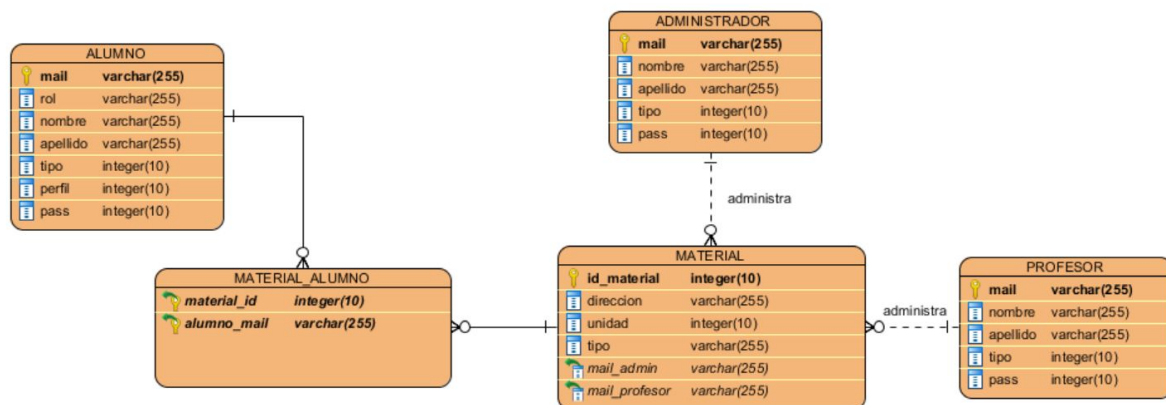
## Entregable 2

Diagrama de secuencia de los componentes del sistema del caso de uso Responder Encuesta



# MODELO RELACIONAL DE LA BASE DE DATOS

Se arregló el modelo relacional obteniendo el siguiente:





# MODELO DE CLASES

Se escogió el patrón de diseño Page Controller junto a MVC para el proyecto, ya que es el que mejor se adecuaba al objetivo de este.

