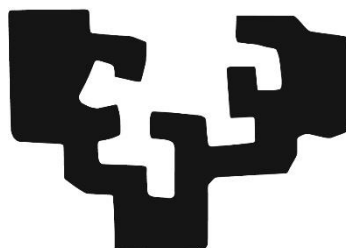


eman ta zabal zazu



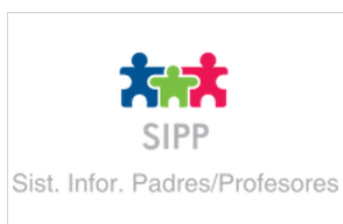
Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

Grado en Informática de Gestión y Sistemas de Información

Trabajo Fin de Grado 2015/2016

SIPP – Sistema de Información entre Padres y Profesores



Alumna: Sandra López Valiente

Directora: Begoña Blanco Jauregi

RESUMEN

Hoy en día en el sector de la educación existe una gran cantidad de sistemas que permiten a los padres tener constancia del seguimiento escolar de sus hijos. Tras hacer una evaluación de las diferentes herramientas y consultar las necesidades de algunos padres, se nos plantea la oportunidad de desarrollar una aplicación más completa y dirigida principalmente a padres con poco tiempo que en su sólo clic pueden saber en todo momento el seguimiento de sus hijos en el colegio y realizar gestiones sin necesidad de papeleo gracias a la utilización de la firma electrónica a través del DNI.

Dado que va ser un sistema destinado a usuarios básicos, se plantea un diseño fácil, intuitivo y que pueda ser utilizado desde cualquier tipo de dispositivo, para que no sólo sea necesario usarlo desde un ordenador. Para ello, se utilizará HTML5 y PHP entre los principales lenguajes de este proyecto.

CONTENIDO

RESUMEN	2
INDICE ILUSTRACIONES	5
INDICE TABLAS	7
1. INTRODUCCIÓN	10
1.1 Propósito	10
1.2 Ámbito	10
1.3 Contexto de negocio	11
1.4 Definición, acrónimos y abreviaturas	12
2. DESCRIPCIÓN DEL PROYECTO	13
3. OBJETIVOS DEL PROYECTO	14
4. ARQUITECTURA	15
5. HERRAMIENTAS	16
6. ALCANCE DEL PROYECTO	18
6.1 Fases del proyecto	18
6.2 Estructura de Descomposición del trabajo EDT	20
6.3 Tareas	22
7. PLANIFICACIÓN TEMPORAL	48
7.1 PERT – CPM	48
7.2 GANTT	50
8. RIESGOS	58
9. EVALUACIÓN ECONÓMICA	61
9.1 Inversión inicial	61
9.2 Recuperación de la inversión	63
10. LEY DE PROTECCIÓN DE DATOS	64
10.1 Introducción	64
10.2 Aplicación de la Ley en nuestro Sistema	64
10.3 Dificultades de la implementación en nuestro Sistema	66
11. DNI-E	68
11.1 Introducción	68
11.2 Necesidades	69
11.3 Uso del DNI en nuestro Sistema	70
11.4 Dificultades de la implementación en nuestro Sistema	73
12. UNIFICACIÓN DEL DESARROLLO	74

12.1	Captura de requisitos	74
12.1.1	Requisitos del Administrador	74
12.1.2	Requisitos del Profesor	74
12.1.3	Requisitos del Padre	75
12.2	Casos de uso	77
12.2.1	Casos de uso del administrador	79
12.2.2	Casos de uso del profesor	84
12.2.3	Casos de uso del tutor	89
12.2.4	Casos de uso del padre.....	94
12.2.5	Casos de uso del usuario	99
12.3	Modelo de Dominio	101
12.4	Diagrama de Clases	102
12.5	Diagrama Entidad Relación Base de Datos	103
12.5.1	Diagramas de secuencia del Administrador.....	104
12.5.2	Diagramas de secuencia de Profesor	112
12.5.3	Diagramas de secuencia de Tutor	124
12.5.4	Diagramas de secuencia de Padre.....	129
12.5.5	Diagramas de secuencia de usuario.....	135
12.6	Plan de Pruebas.....	137
12.6.1	Plan de pruebas usuario.....	137
12.6.2	Plan de pruebas administrador	139
12.6.3	Plan de pruebas profesor	141
12.6.4	Plan de pruebas tutor.....	144
12.6.5	Plan de pruebas padre	146
13.	ESFUERZO PLANIFICADO VS. ESFUERZO REAL	148
14.	POSIBLES MEJORAS	150
15.	CONCLUSIONES	151
16.	BIBLIOGRAFÍA.....	152

INDICE ILUSTRACIONES

Ilustración 1 – EDT Parte 1	20
Ilustración 2 – EDT Parte 2	21
Ilustración 3 - Gantt 1	50
Ilustración 4 - Gantt 2	51
Ilustración 5 - Gantt 3	52
Ilustración 6 - Gantt 4	53
Ilustración 7 - Gantt 5	54
Ilustración 8 - Gantt 6	55
Ilustración 9 - Gantt 7	56
Ilustración 10 - Gantt 8	57
Ilustración 11 - Casos de uso	77
Ilustración 12 - Jerarquía actores	78
Ilustración 13 - Modelo de Dominio	101
Ilustración 14 - Diagrama de Clases	102
Ilustración 15 - Diagrama Entidad-Relación	103
Ilustración 16 Diagrama Gestionar Alumno	104
Ilustración 17 - Diagrama Añadir alumno	105
Ilustración 18 - Diagrama Editar Alumno	106
Ilustración 19 - Diagrama Eliminar Alumno	107
Ilustración 20 - Diagrama Gestionar Circulares	108
Ilustración 21 - Diagrama Añadir Circular	109
Ilustración 22 - Diagrama Editar Circular	110
Ilustración 23 - Diagrama Eliminar Circular	111
Ilustración 24 - Diagrama Gestionar Deberes	112
Ilustración 25 - Diagrama Añadir Deber	113
Ilustración 26 - Diagrama Editar Deber	114
Ilustración 27 - Diagrama Eliminar Deber	115
Ilustración 28 - Diagrama Eliminar Deber	116
Ilustración 29 - Diagrama Gestionar Exámenes	117
Ilustración 30 - Diagrama Añadir Examen	118

Ilustración 31 - Diagrama Editar Examen	119
Ilustración 32 - Diagrama Eliminar Examen.....	120
Ilustración 33 - Diagrama Gestionar Faltas	121
Ilustración 34 - Diagrama Añadir Falta	122
Ilustración 35 - Diagrama Ver Faltas.....	123
Ilustración 36 - Diagrama Gestionar Citas	124
Ilustración 37 - Diagrama Eliminar Cita	125
Ilustración 38 - Diagrama Ver Cita.....	126
Ilustración 39 - Diagrama Gestionar Circulares	127
Ilustración 40 - Diagrama Preparar notas.....	128
Ilustración 41 - Diagrama Ver Información Hijo	129
Ilustración 42 - Diagrama Enviar Mensaje.....	130
Ilustración 43 - Diagrama Ver Notas.....	131
Ilustración 44 - Diagrama Ver Deberes.....	132
Ilustración 45 - Diagrama Ver Exámenes.....	133
Ilustración 46 - Diagrama Ver Mensajes.....	134
Ilustración 47 - Diagrama Identificarse	135
Ilustración 48 - Diagrama Recuperar Contraseña	136
Ilustración 49 - Gráfica horas planificadas vs horas reales	149

INDICE TABLAS

Tabla 1 – Tarea 1.1	22
Tabla 2 – Tarea 1.2	23
Tabla 3 – Tarea 1.3	24
Tabla 4 – Tarea 1.4	25
Tabla 5 – Tarea 1.5	26
Tabla 6 – Tarea 2.1	27
Tabla 7 – Tarea 2.2	28
Tabla 8 – Tarea 2.3	29
Tabla 9 – Tarea 2.4	30
Tabla 10 – Tarea 2.5	31
Tabla 11 – Tarea 3.1	32
Tabla 12 – Tarea 3.2	33
Tabla 13 – Tarea 4.1	34
Tabla 14 – Tarea 4.2	35
Tabla 15 – Tarea 5.1	36
Tabla 16 – Tarea 5.2	37
Tabla 17 – Tarea 5.3	38
Tabla 18 – Tarea 5.4	39
Tabla 19 – Tarea 5.5	40
Tabla 20 – Tarea 6.1	41
Tabla 21 – Tarea 6.2	42
Tabla 22 – Tarea 6.3	43
Tabla 23 – Tarea 6.4	44
Tabla 24 - Tarea 7.1	45
Tabla 25 - Tarea 7.2	46
Tabla 26 - Tarea 8.1	47
Tabla 27 – PERT / CPM	49
Tabla 28 - Riesgo 1.....	58
Tabla 29 - Riesgo 2.....	59
Tabla 30 - Riesgo 3.....	59

Tabla 31 - Riesgo 4.....	60
Tabla 32 - Riesgo 5.....	60
Tabla 33 - Inversión proyecto.....	63
Tabla 34 - Caso de uso Administrador - Añadir Alumno	79
Tabla 35 - Caso de uso Administrador - Editar Alumno	80
Tabla 36 - Caso de uso Administrado - Eliminar Alumno	80
Tabla 37 - Caso de uso Administrador - Añadir salida00.....	81
Tabla 38 - Caso de uso Administrador - Editar salida.....	81
Tabla 39 - Caso de uso Administrador - Eliminar salida	82
Tabla 40 - Caso de uso Administrador - Añadir usuario.....	82
Tabla 41 - Caso de uso Administrador - Editar usuario	83
Tabla 42 - Caso de uso Administrador - Eliminar usuario	83
Tabla 43 - Caso de uso Profesor - Enviar mensaje	84
Tabla 44 - Caso de uso Profesor - Añadir deber.....	84
Tabla 45 – Caso de uso Profesor – Editar deber.....	85
Tabla 46 - Caso de uso Profesor - Eliminar Deber	85
Tabla 47 - Caso de uso Profesor - Añadir examen.....	86
Tabla 48 - Caso de uso Profesor - Editar examen.....	86
Tabla 49 - Caso de uso Profesor - Eliminar examen	87
Tabla 50 - Caso de uso Profesor - Añadir falta	87
Tabla 51 - Caso de uso Profesor - Ver faltas.....	88
Tabla 52 - Caso de uso Tutor - Enviar Circular.....	89
Tabla 53 - Caso de uso Tutor - Añadir hora	89
Tabla 54 - Caso de uso Tutor - Eliminar cita	90
Tabla 55 - Caso de uso Tutor - Eliminar hora	90
Tabla 56 - Caso de uso Tutor - Ver citas	91
Tabla 57 - Caso de uso Tutor - Preparar notas	92
Tabla 58 - Caso de uso tutor - Ver respuesta	93
Tabla 59 - Caso de uso Padre - Enviar mensaje.....	94
Tabla 60 - Caso de uso Padre – Firmar circulares.....	94
Tabla 61 - Caso de uso Padre - Firmar notas	95
Tabla 62 - Caso de uso Padre - Editar datos hijo	96

Tabla 63 - Caso de uso Padre - Firmar justificación.....	97
Tabla 64 - Caso de uso Padre - Ver deber	97
Tabla 65 - Caso de uso Padre - Ver examen	98
Tabla 66 - Caso de uso Padre - Ver mensajes.....	98
Tabla 67 - Caso de uso Usuario - Identificarse	99
Tabla 68 - Caso de uso Usuario - Recuperar contraseña.....	100
Tabla 69 - Pruebas Usuario	139
Tabla 70 - Pruebas Administrador	140
Tabla 71 - Pruebas profesor	143
Tabla 72 - Pruebas Tutor	145
Tabla 73 - Pruebas Padre	147
Tabla 74 - Esfuerzo planificado vs esfuerzo real	149
Tabla 75 - Comparación horas planificadas vs reales.....	149

1. INTRODUCCIÓN

En este proyecto se realiza el desarrollo de una aplicación que sirva de comunicación para colegios de educación obligatoria entre los profesores y los padres de sus alumnos.

1.1 Propósito

El resultado final será la creación de un sistema que permita a los colegios gestionar los datos de sus alumnos y que a su vez los padres estén en conocimiento constante del seguimiento de sus hijos en cada asignatura del curso, pudiendo interactuar con los profesores online.

Sólo los usuarios de esta aplicación podrán tener acceso a la misma, los profesores serán los encargados de mantener la información de los alumnos de las asignaturas que imparte, podrá definir los deberes, exámenes, añadir faltas de asistencia, mandar mensajes a los padres y poner las notas de cada trimestre. Sólo los tutores del curso podrán concertar citas con los padres y enviar circulares. Estos podrán consultar todos los datos de sus hijos, podrán justificar las faltas y firmar las circulares mandadas por los tutores y notas,

El administrador será el encargado de gestionar tanto la información del centro, como de dar de alta a los alumnos, padres y profesores.

1.2 Ámbito

El proyecto se basa en realizar una aplicación web que sirva tanto a los profesores como a los padres el manejo de la misma a través de cualquier dispositivo.

1.3 Contexto de negocio

Esta aplicación va dirigida al mercado educativo, en concreto a colegios de educación obligatoria, entre la educación primaria y secundaria. Hoy en día, existen numerosas aplicaciones en este ámbito, aplicaciones en las que el usuario puede conocer los avances de su hijo durante el curso académico, sin embargo, este sistema va más allá, permite un control total al instante de lo que está haciendo su hijo. Pudiendo mantener comunicación directa con los profesores.

Hoy en día, existen una gran cantidad de aplicaciones en el mundo de la educación, para facilitar la vida tanto de los trabajadores de un centro como de lo padres de sus alumnos. Sin embargo, con esta aplicación lo que se pretende es mejorar la calidad del servicio y dar tanto a los padres como a los docentes más facilidades y de una manera segura gracias a la firma digital.

Se decide basar esta aplicación en un entorno web HTML5, ya que es una herramienta que permite ejecutarse desde la mayoría de los sistemas operativos y navegadores, pudiendo ejecutarse desde cualquier dispositivo sin la necesidad de hacer varias aplicaciones en función del dispositivo en el que ejecutarse, permitiendo la diversidad de conexión.

1.4 Definición, acrónimos y abreviaturas

- **SIPP:** Sistema de información entre Padres y Profesores.
- **DNI:** Documento Nacional de Identidad

2. DESCRIPCIÓN DEL PROYECTO

Se desarrollará un sistema denominado SIPP, una aplicación que permitirá controlar y gestionar los datos de los alumnos de un colegio, en todas sus asignaturas. Para responder a las necesidades del colegio el sistema deberá incluir un apartado en el que los administradores se encargarán de dar de alta a los alumnos, información del centro y circulares que serán gestionadas desde dirección.

Los profesores serán los encargados de rellenar los datos de sus alumnos que estén matriculados en las asignaturas que imparten, tanto exámenes, deberes, faltas, notas y mensajes. Sólo los tutores podrán concertar citas y enviar las circulares gestionadas por el centro a los padres de los alumnos del curso del que es tutor.

Asimismo, los padres podrán visualizar todos los datos que los profesores han gestionado sobre sus hijos, podrá enviar mensajes a cualquiera de los profesores. Sin embargo, al igual que, al contrario, sólo podrá concertar citas con el profesor que sea el tutor.

Por último, para que se certifique la entrega de notas, justificación de faltas y circulares, los padres tendrán que firmar con el DNI electrónico.

3. OBJETIVOS DEL PROYECTO

El objetivo principal del proyecto es desarrollar un sistema que permita a los colegios la gestión de los datos de los alumnos y facilite la comunicación de profesor-padre de manera inmediata.

Se desea que los padres tengan un conocimiento diario, de lo que sus hijos tienen que hacer, cómo van en sus tareas, evitando que el alumno sea el intermediario entre los padres y profesores y de una manera prácticamente inmediata, sin necesidad de tener que contactar con el colegio esperando respuesta.

Asimismo, la aplicación mantendrá un aspecto claro, fácil e intuitivo para el manejo del usuario, siendo conscientes de que se desconoce las habilidades informáticas del usuario.

4. ARQUITECTURA

La arquitectura a implementar que hemos escogido es la de Cliente/Servidor, ya que en nuestro proyecto va ser una aplicación donde el cliente va hacer una petición a nuestra aplicación y nosotros (el servidor) les vamos a dar servicio.



5. HERRAMIENTAS

A continuación, se detallan las diferentes herramientas que se han utilizado para la elaboración del proyecto:

- **Pencil:** Para realizar los borradores de las interfaces, hemos elegido esta herramienta, ya que es fácil, intuitiva y no requiere ningún registro extra más que la cuenta de correo.



- **Dia:** Es un programa que realiza todo tipo de diagramas, es muy completo, abarca todo tipo de diagramas y es muy fácil de usar.



- **WBS Tool:** Es una herramienta web que nos permite realizar EDT (), la elegimos dado que es online, sólo necesitamos el email para poder utilizarlo, puedes realizar diferentes diagramas y almacenarlos en la nube y en el momento de necesitarlo descargártelo como imagen o imprimirlo.



- **Sublime text:** Con este editor de texto vamos a programar nuestra aplicación ya que sirve tanto para programar en html5, php, css y js.



- **MySQL:** Lo vamos utilizar para diseñar la base de datos, ya que también estamos más familiarizado con él y su introducción en eclipse a la hora de programar.



- **XAMPP:** Es el servidor independiente, de software libre, donde vamos alojar nuestra aplicación, ya que es gratuito, fácil de utilizar, lo hemos utilizado durante toda la carrera para realizar diferentes proyectos y nos permite usarlo sin ningún tipo de desconocimiento. Es muy completo, nos sirve para la gestión de bases de datos mysql, servidor web Apache e intérpretes para lenguajes php y perl.



- **Office:** Vamos a utilizarlo para la realización de la documentación necesaria durante el desarrollo del proyecto. Lo hemos elegido porque es el conjunto de herramientas de productividad más conocido y del que más conocimiento se dispone.



- **Smartsheet:** Herramienta online que permite la realización de diagramas GANTT, es fácil, intuitivo y aunque no es gratuito, tiene una versión de prueba que para lo que se necesita en este proyecto, nos es muy útil.



6. ALCANCE DEL PROYECTO

En este punto se realizará la descomposición del trabajo necesario, representativo de la totalidad del mismo, para dar por terminado un proyecto.

6.1 Fases del proyecto

Especificación de necesidades: Durante esta fase, se dejarán claras las necesidades que queremos cubrir en nuestra aplicación, se buscarán aplicaciones similares para poder hacerla lo más completa posible y mejorar lo que ya existe si es posible y crearemos el proyecto de viabilidad.

Análisis: Con la información necesaria se deberá concretar cómo se va a llevar a cabo el proyecto de la mejor manera posible. Para ello se realizará la captura de requisitos completándola con la ayuda de la primera reunión con la directora del proyecto y obtendremos las distintas funcionalidades de nuestro sistema. Además, será el momento de decidir cuáles serán las herramientas a utilizar para la implementación.

Diseño: En esta fase nos dedicaremos a diseñar los distintos elementos que serán necesarios a lo largo de la implementación. Daremos forma a lo que será nuestra implementación final a través de los distintos diagramas.

Aprendizaje: En esta fase, nos centraremos en realizar búsqueda de información y manuales de las herramientas que vamos a utilizar, para podernos adentrar en la implementación con los conceptos claros y estudiados.

Implementación: En esta fase se realizará la transformación en código del diseño obtenido en los puntos anteriores. Con la información obtenida nos servirá para plasmar nuestro desarrollo en real.

Será la fase a la que más tiempo se le va a dedicar, ya que, al realizar el desarrollo de una manera real, nos surgirán dudas que deberemos ir corrigiendo a medida que vamos avanzando.

Pruebas: Durante la fase de implantación se van a realizar pruebas que servirán para comprobar que vamos por el buen camino, sin embargo, en esta etapa se realizará las pruebas más críticas, se tratará de ‘romper el desarrollo’ como si fuésemos el usuario final, para poder contemplar todas las posibilidades y no dejar margen de error.

Documentación: Fase orientada a la creación de los manuales para los diferentes usuarios de la aplicación (administradores, profesores y padres), además realizaremos la memoria del proyecto que agrupará toda la información obtenida en las anteriores fases.

Cierre: Fase de conclusión del proyecto, empaquetaremos toda la información a entregar a procederemos a entregársela a la persona correspondiente.

6.2 Estructura de Descomposición del trabajo EDT

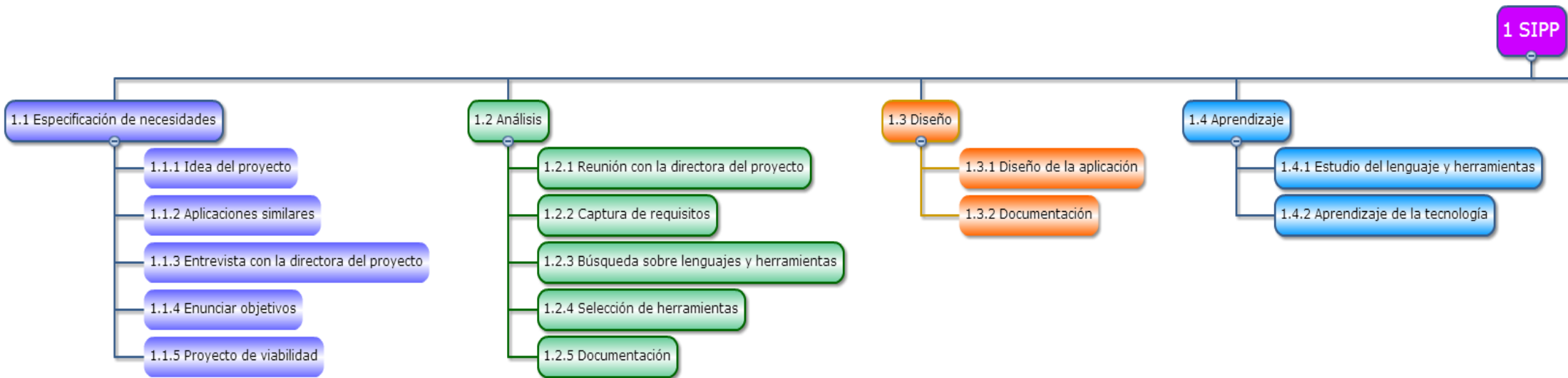


Ilustración 1 – EDT Parte 1

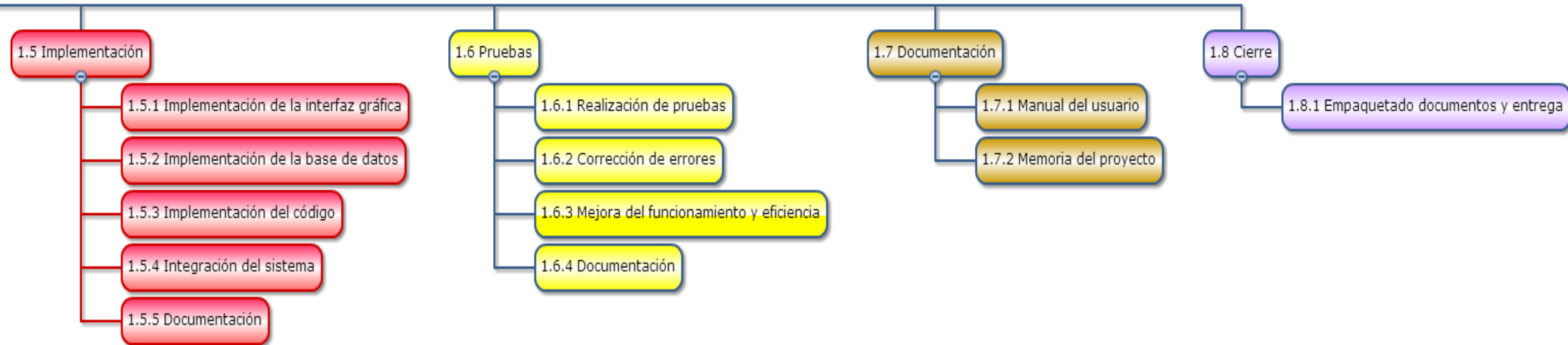


Ilustración 2 – EDT Parte 2

6.3 Tareas

1-Especificación de necesidades

Paquete de Trabajo 1.1: **Idea del proyecto**

Duración: **10 horas**

Tabla 1 – Tarea 1.1

Descripción:

Búsqueda de información para conseguir una única idea de proyecto lo más completa posible, definiéndola para poder exponerla a la directora del proyecto.

Entradas:

Ninguna.

Salidas/Entregables:

Información recopilada.

Recursos necesarios:

Herramientas de búsqueda de internet.

Precedencias:

Ninguna ya que se parte de cero.

1-Especificación de necesidades

Paquete de Trabajo 1.2: **Aplicaciones similares**

Duración: **2 horas.**

Tabla 2 – Tarea 1.2

Descripción:

Una vez que se tiene concretada la idea a desarrollar, se buscan aplicaciones similares que se encuentran en el mercado, para detectar necesidades e intentar mejorar lo que ya existe.

Entradas:

Ninguna.

Salidas/Entregables:

Información sobre lo que existe en el mercado.

Recursos necesarios:

Herramientas de búsqueda de internet.

Precedencias:

1.1

1-Especificación de necesidades

Paquete de Trabajo 1.3: **Entrevista con la directora del proyecto**

Duración: **2 horas.**

Tabla 3 – Tarea 1.3

Descripción:

Se queda con la directora del proyecto para exponer las ideas y establecer los objetivos que se quieren conseguir con la realización del mismo.

Entradas:

Ninguna.

Salidas/Entregables:

Objetivos del proyecto.

Recursos necesarios:

Lugar para mantener la reunión e información obtenida hasta el momento.

Precedencias:

1.1 y 1.2

1-Especificación de necesidades

Paquete de Trabajo 1.4: **Enunciar objetivos**

Duración: **2 horas.**

Tabla 4 – Tarea 1.4

Descripción:

Se trata de enunciar los objetivos a conseguir con el proyecto de una forma clara y precisa.

Entradas:

Información recopilada de la entrevista con la directora del proyecto.

Salidas/Entregables:

Parte del Documento de Objetivos del Proyecto, con los objetivos bien documentados.

Recursos necesarios:

Nos hace falta saber de forma clara y precisa lo que nos pide la directora del proyecto.

Precedencias:

1.1, 1.2 y 1.3

1-Especificación de necesidades

Paquete de Trabajo 1.5: **Proyecto de viabilidad**

Duración: **25 horas.**

Tabla 5 – Tarea 1.5

Descripción:

Se trata de documentar cuál es el trabajo que se va a desarrollar. Nos ayudará a planificar todo el desarrollo del proyecto. Este documento incluirá una serie de puntos, como puede ser: Objetivos del proyecto, Descripción general, Alcance del proyecto, calendario y estructuración del trabajo, Análisis de riesgos, Viabilidad económica.

Entradas:

Ninguna.

Salidas/Entregables:

Documento proyecto de viabilidad.

Recursos necesarios:

Ordenador.

Precedencias:

1.1, 1.2, 1.3 y 1.4

2. Análisis

Paquete de Trabajo 2.1: **Reunión con la directora del proyecto**

Duración: **2 horas.**

Tabla 6 – Tarea 2.1

Descripción:

Realizar una reunión para concretar en profundidad las necesidades del producto. Se intentará dejar claro que los objetivos son los que queremos conseguir con el resultado final.

Entradas:

Entrevista previa.

Salidas/Entregables:

Listado de funcionalidades de la aplicación.

Recursos necesarios:

Lugar para mantener la reunión e información obtenida hasta el momento.

Precedencias:

1.3

2. Análisis

Paquete de Trabajo 2.2: **Captura de requisitos**

Duración: **7 horas.**

Tabla 7 – Tarea 2.2

Descripción:

En esta tarea y gracias a la información obtenida en las reuniones con la directora, se obtienen los requisitos de nuestra aplicación. Se debe indicar el comportamiento que tendrá la aplicación.

Entradas:

Información de las reuniones y los objetivos documentados.

Salidas/Entregables:

Modelo de casos de uso y Modelo de dominio.

Recursos necesarios:

Ordenador.

Precedencias:

2.2

2-Análisis

Paquete de Trabajo 2.3: **Búsqueda de bibliografía sobre lenguajes y herramientas**

Duración: **25 horas.**

Tabla 8 – Tarea 2.3

Descripción:

Durante esta fase habrá que buscar la mayor cantidad de bibliografía seleccionando la que realmente necesitamos basándonos en información útil, completa y que la fuente de procedencia sea segura.

Entradas:

Ninguna.

Salidas/Entregables:

Información con toda la bibliografía recogida.

Recursos necesarios:

Ordenador e internet.

Precedencias:

2.2

2. Análisis

Paquete de Trabajo 2.4: **Selección de herramientas**

Duración: **3 horas.**

Tabla 9 – Tarea 2.4

Descripción:

En esta tarea se eligen, según nuestras necesidades, las herramientas que vamos a utilizar para desarrollar nuestro proyecto.

Entradas:

Información con las diversas herramientas posibles a utilizar.

Salidas/Entregables:

Información con las herramientas a utilizar.

Recursos necesarios:

Ordenador e internet.

Precedencias:

2.3

2- Análisis

Paquete de Trabajo 2.5: **Documentación**

Duración: **10 horas**

Tabla 10 – Tarea 2.5

Descripción:

Se trata de documentar todos los puntos de análisis comentado.

Entradas:

Toda la información obtenida durante el análisis.

Salidas/Entregables:

Documentación fase de Análisis, Modelo de Casos de Uso, Modelo de Casos de Uso Extendido y Modelo de Dominio.

Recursos necesarios:

Word y Dia.

Precedencias:

2.1, 2.2, 2.3 y 2.4

3- Diseño

Paquete de Trabajo 3.1: **Diseño de la aplicación**

Duración: **35 horas**

Tabla 11 – Tarea 3.1

Descripción:

Fase en la que se diseña la estructura sobre la que se programará la aplicación, encontrando la forma del sistema que cumpla con todos los requisitos. Es decir, un esquema con las diferentes pantallas que compondrán la aplicación, teniendo en cuenta las funciones a realizar en cada una de ellas.

Entradas:

Información de los procesos a desarrollar del análisis.

Salidas/Entregables:

Bocetos de diseño.

Recursos necesarios:

Ordenador

Precedencias:

2.2

3- Diseño

Paquete de Trabajo 3.2: **Documentación**

Duración: **12 horas**

Tabla 12 – Tarea 3.2

Descripción:

Se trata de realizar la documentación de los puntos del diseño.

Entradas:

Todos los puntos previos.

Salidas/Entregables:

Documento en el que va detallado la fase del diseño, Diagramas de Secuencia y Diagrama de Clases.

Recursos necesarios:

Word y Dia.

Precedencias:

3.1

4- Aprendizaje

Paquete de Trabajo 4.1: Estudio de lenguaje y herramientas

Duración: **40 horas**

Tabla 13 – Tarea 4.1

Descripción:

Estudiar y aprender en profundidad conocimientos de HTML5, php y javascript.

Entradas:

Bibliografía.

Salidas/Entregables:

Conocimientos adquiridos.

Recursos necesarios:

Ordenador, internet y manuales.

Precedencias:

2.3

4- Aprendizaje

Paquete de Trabajo 4.2: **Aprendizaje de la tecnología**

Duración: **8 horas**

Tabla 14 – Tarea 4.2

Descripción:

Aprender a utilizar los conocimientos adquiridos en el punto anterior.

Entradas:

Herramientas seleccionadas.

Salidas/Entregables:

Información sobre las herramientas.

Recursos necesarios:

XAMPP, Sublime Text, tutoriales y ejercicios prácticos.

Precedencias:

2.3 y 2.4

5- Implementación

Paquete de Trabajo 5.1: **Implementación de la interfaz gráfica**

Duración: **30 horas**

Tabla 15 – Tarea 5.1

Descripción:

Fase en la que se implementan todas las interfaces necesarias.

Entradas:

Diseño de la aplicación.

Salidas/Entregables:

Interfaces.

Recursos necesarios:

XAMPP y Sublime Text.

Precedencias:

3.1 y 4.2

5- Implementación

Paquete de Trabajo 5.2: **Implementación de la base de datos**

Duración: **20 horas**

Tabla 16 – Tarea 5.2

Descripción:

Implementación de la base de datos que va utilizar el sistema.

Entradas:

Diseño de la aplicación.

Salidas/Entregables:

Tablas de la base de datos.

Recursos necesarios:

PhpMyAdmin

Precedencias:

3.1, 4.2 y 5.1

5- Implementación

Paquete de Trabajo 5.3: **Implementación del código**

Duración: **170 horas**

Tabla 17 – Tarea 5.3

Descripción:

Implementar el código del sistema.

Entradas:

La base de datos y diseño de la implementación.

Salidas/Entregables:

Implementación del sistema.

Recursos necesarios:

XAMPP, Sublime Text y PhpMyAdmin.

Precedencias:

3.1, 4.2, 5.1 y 5.2

5- Implementación

Paquete de Trabajo 5.4: **Integración del sistema**

Duración: **12 horas**

Tabla 18 – Tarea 5.4

Descripción:

En esta fase se le da el último retoque visual y se integra todo.

Entradas:

Aplicación.

Salidas/Entregables:

Aplicación mejorada.

Recursos necesarios:

XAMPP, Sublime Text y PhpMyAdmin.

Precedencias:

3.1, 4.2, 5.1, 5.2 y 5.3

5- Implementación

Paquete de Trabajo 5.5: **Documentación**

Duración: **7 horas**

Tabla 19 – Tarea 5.5

Descripción:

Realizar documentación de toda la implementación.

Entradas:

Toda la fase de la implementación.

Salidas/Entregables:

Documento en el que se plasmará toda la implementación del sistema.

Recursos necesarios:

Word.

Precedencias:

5.1, 5.2, 5.3 y 5.4

6- Pruebas

Paquete de Trabajo 6.1: **Realización de pruebas**

Duración: **8 horas**

Tabla 20 – Tarea 6.1

Descripción:

Comprobar que la aplicación cumple con todo lo recogido en la tarea de Captura de requisitos. Observando que funcione bien y buscando posibles fallos.

Entradas:

Aplicación.

Salidas/Entregables:

Aplicación probada.

Recursos necesarios:

XAMPP.

Precedencias:

2.2, 3 y 5

6- Pruebas

Paquete de Trabajo 6.2: **Corrección de errores**

Duración: **16 horas**

Tabla 21 – Tarea 6.2

Descripción:

Se corrigen todos los errores encontrados para que la aplicación cumpla con todos los requisitos.

Entradas:

Aplicación.

Salidas/Entregables:

Aplicación sin errores.

Recursos necesarios:

XAMPP, Sublime Text y PhpMyAdmin.

Precedencias:

2.2 y 6.1

6- Pruebas

Paquete de Trabajo 6.3: **Mejora del funcionamiento y eficiencia**

Duración: **7 horas**

Tabla 22 – Tarea 6.3

Descripción:

Se revisa el código y se intenta hacer más eficiente en la medida de lo posible.

Entradas:

Aplicación sin errores.

Salidas/Entregables:

Aplicación final.

Recursos necesarios:

XAMPP, Sublime Text y PhpMyAdmin.

Precedencias:

2.2, 5, 6.1 y 6.2

6- Pruebas

Paquete de Trabajo 6.4: **Documentación**

Duración: **5 horas**

Tabla 23 – Tarea 6.4

Descripción:

Realizar documentación de todas las fases de las pruebas.

Entradas:

Información de todos los paquetes de prueba.

Salidas/Entregables:

Documentos con las pruebas detalladas y explicación de los errores.

Recursos necesarios:

Word.

Precedencias:

6.1, 6.2 y 6.3

7- Documentación

Paquete de Trabajo 7.1: **Manual del usuario**

Duración: **5 horas**

Tabla 24 - Tarea 7.1

Descripción:

Realizar documento que sirva de manual para los usuarios que van a utilizar la aplicación, debe estar explicado de manera fácil e intuitiva para que estos sin tener conocimientos de informática sean capaces de utilizarlo.

Entradas:

Aplicación final.

Salidas/Entregables:

Manual de usuario.

Recursos necesarios:

Word.

Precedencias:

3, 5 y 6

7- Documentación

Paquete de Trabajo 7.2 **Memoria del proyecto**

Duración: **7 horas**

Tabla 25 - Tarea 7.2

Descripción:

En esta tarea recopilaremos toda la información obtenida en las fases anteriores además de añadir más información relevante acerca del proyecto, redactando una memoria final.

Entradas:

Toda la documentación obtenida en los puntos anteriores.

Salidas/Entregables:

Memoria del proyecto.

Recursos necesarios:

Word.

Precedencias:

1, 2, 3, 5 y 6

8- Cierre

Paquete de Trabajo 8.1: **Empaquetado documentos y entrega**

Duración: **2 horas**

Tabla 26 - Tarea 8.1

Descripción:

Esta es la última tarea, y en ella empaquetaremos todos y procederemos a la entrega del trabajo fin de grado.

Entradas:

Toda la documentación obtenida en los puntos anteriores.

Salidas/Entregables:

Trabajo fin de grado.

Recursos necesarios:

-

Precedencias:

6 y 7

7. PLANIFICACIÓN TEMPORAL

Se detallará la planificación estimada de cada una de las tareas asignadas durante el alcance.

7.1 PERT – CPM

ID TAREA	FASE/TAREA	COMIENZO	FIN	DURACIÓN (HORAS)
1	Especificación de necesidades	02/03/2015	19/03/2015	41
1.1	Idea del proyecto	02/03/2015	05/03/2015	10
1.2	Aplicaciones similares	05/03/2015	05/03/2015	2
1.3	Entrevista con la directora del proyecto	06/03/2015	06/03/2015	2
1.4	Enunciar objetivos	09/03/2015	09/03/2015	2
1.5	Proyecto de viabilidad	09/03/2015	19/03/2015	25
2	Análisis	23/03/2015	13/04/2015	47
2.1	Reunión con la directora del proyecto	23/03/2015	23/03/2015	2
2.2	Captura de requisitos	23/03/2015	25/03/2015	7
2.3	Búsqueda sobre lenguajes y herramientas	26/03/2015	06/04/2015	25
2.4	Selección de herramientas	06/04/2015	07/04/2015	3
2.5	Documentación	08/04/2015	13/04/2015	10
3	Diseño	14/04/2015	06/05/2015	47
3.1	Diseño de la aplicación	14/04/2015	29/04/2015	35
3.2	Documentación	30/04/2015	06/05/2015	12
4	Aprendizaje	08/05/2015	02/06/2015	55
4.1	Estudio del lenguaje y herramientas	08/05/2015	26/05/2015	40
4.2	Aprendizaje de la tecnología	27/05/2015	02/06/2015	15
5	Implementación	03/06/2015	30/10/2015	239
5.1	Implementación de la interfaz gráfica	03/06/2015	16/06/2015	30
5.2	Implementación de la base de datos	17/06/2015	25/06/2015	20
5.3	Implementación del código	25/06/2015	21/10/2015	170
5.4	Integración del sistema	22/10/2015	27/10/2015	12
5.5	Documentación	28/10/2015	30/10/2015	7
6	Pruebas	03/11/2015	18/11/2015	36
6.1	Realización de pruebas	03/11/2015	05/11/2015	8
6.2	Corrección de errores	05/11/2015	12/11/2015	16

6.3	Mejora del funcionamiento y eficiencia	13/11/2015	17/11/2015	7
6.4	Documentación	17/11/2015	18/11/2015	5
7	Documentación	19/11/2015	01/12/2015	12
7.1	Manual de usuario	19/11/2015	24/11/2015	5
7.2	Memoria del proyecto	25/11/2015	01/12/2015	7
8	Cierre	08/02/2016	08/02/2016	2
8.1	Empaquetado documentos y entrega	08/02/2016	08/02/2016	2
TOTAL	Trabajo Fin de Grado	02/03/2015	01/12/2015	533

Tabla 27 – PERT / CPM

7.2 GANTT



Ilustración 3 - Gantt 1

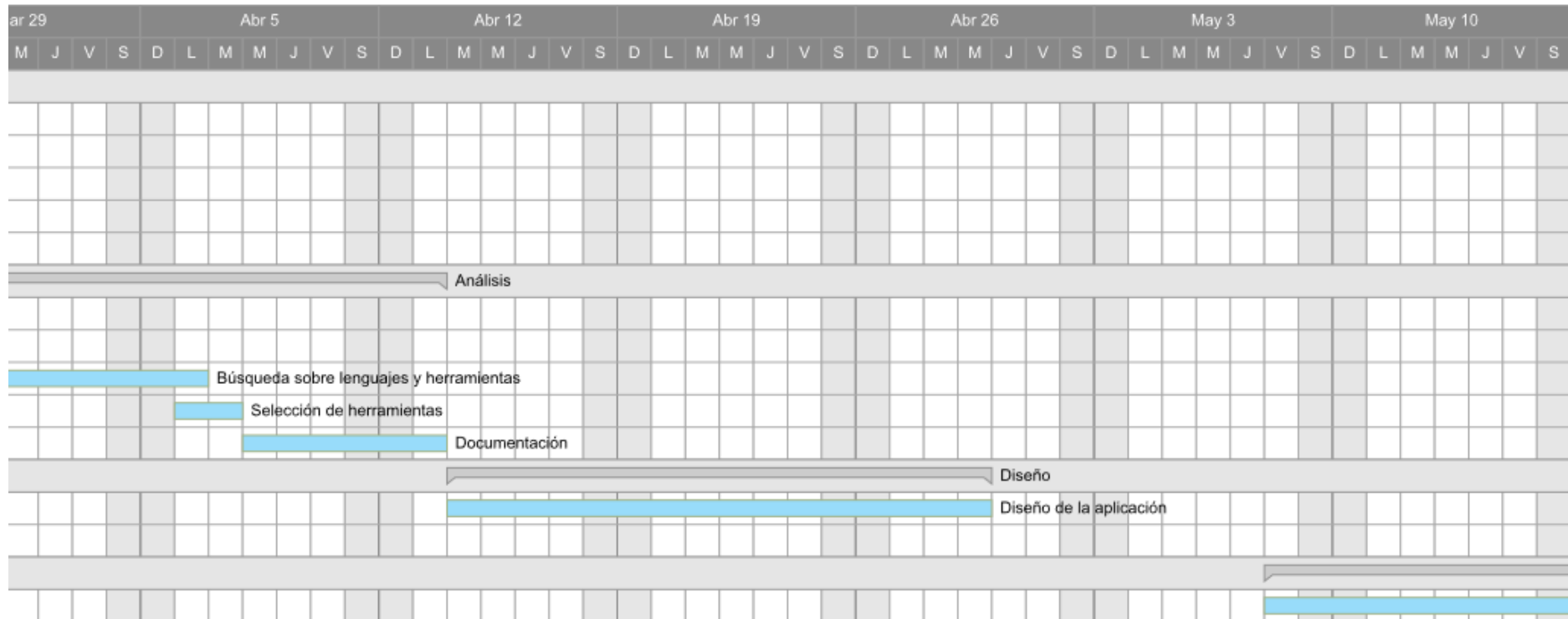
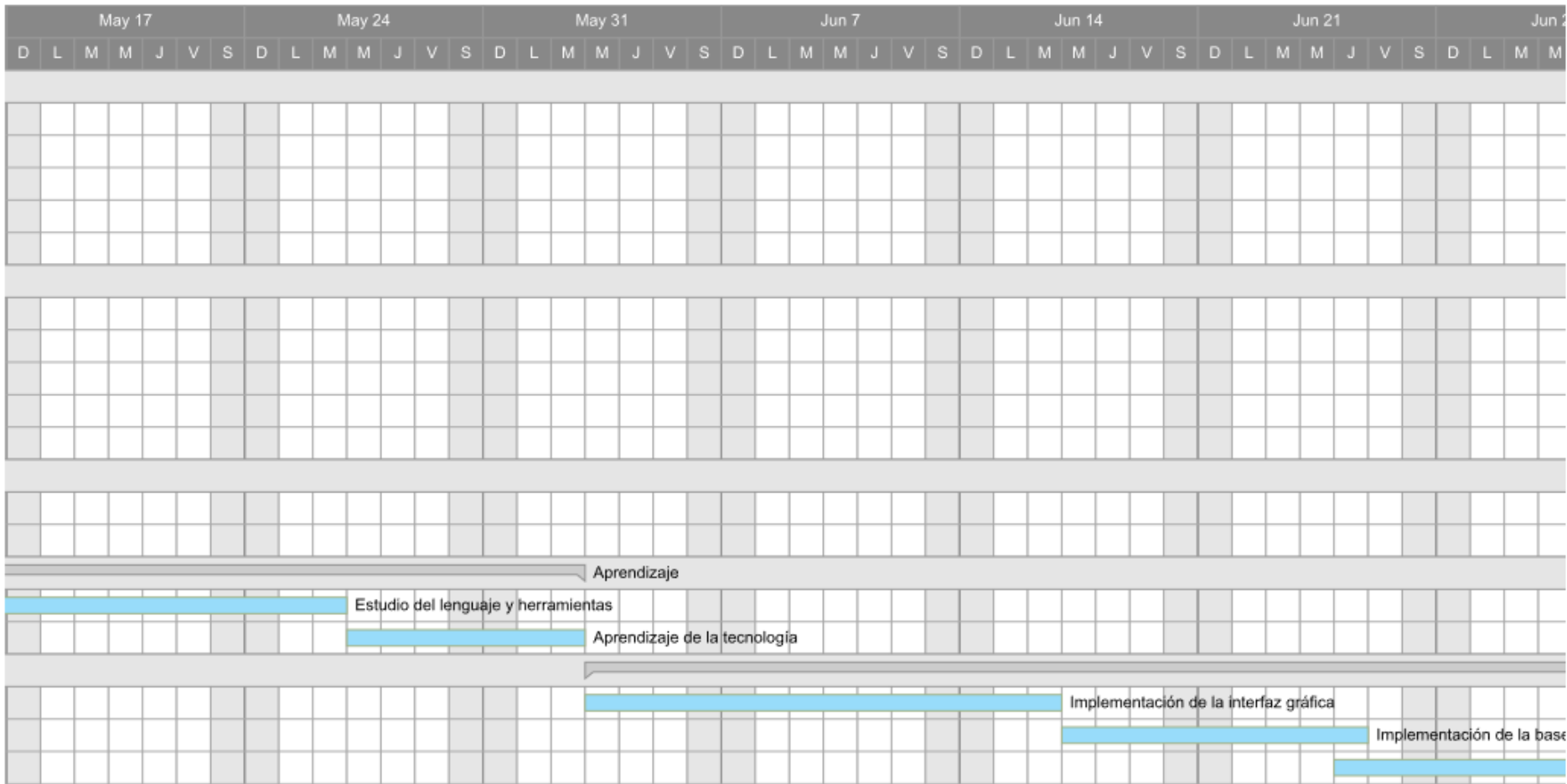


Ilustración 4 - Gantt 2



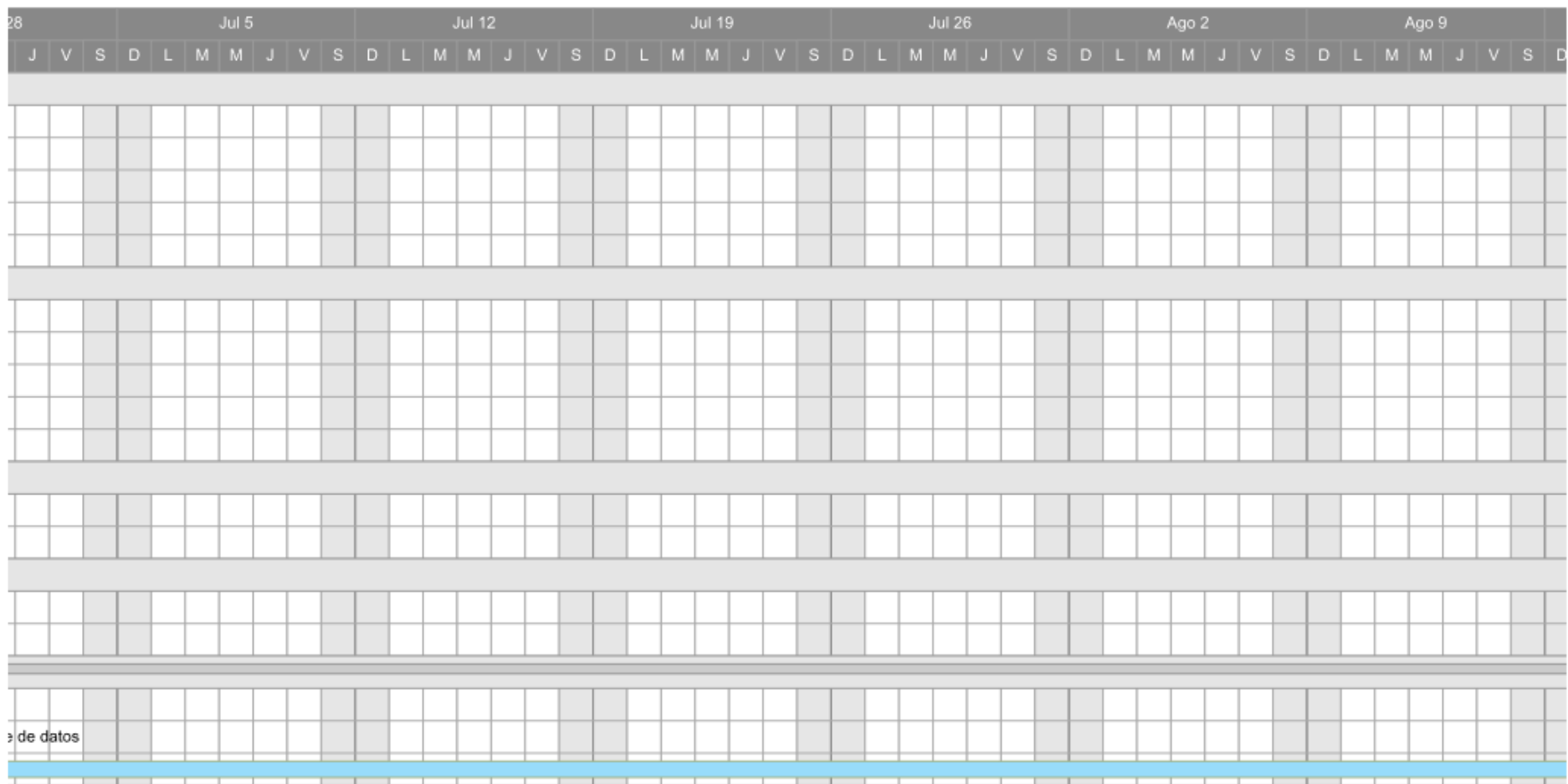


Ilustración 6 - Gantt 4

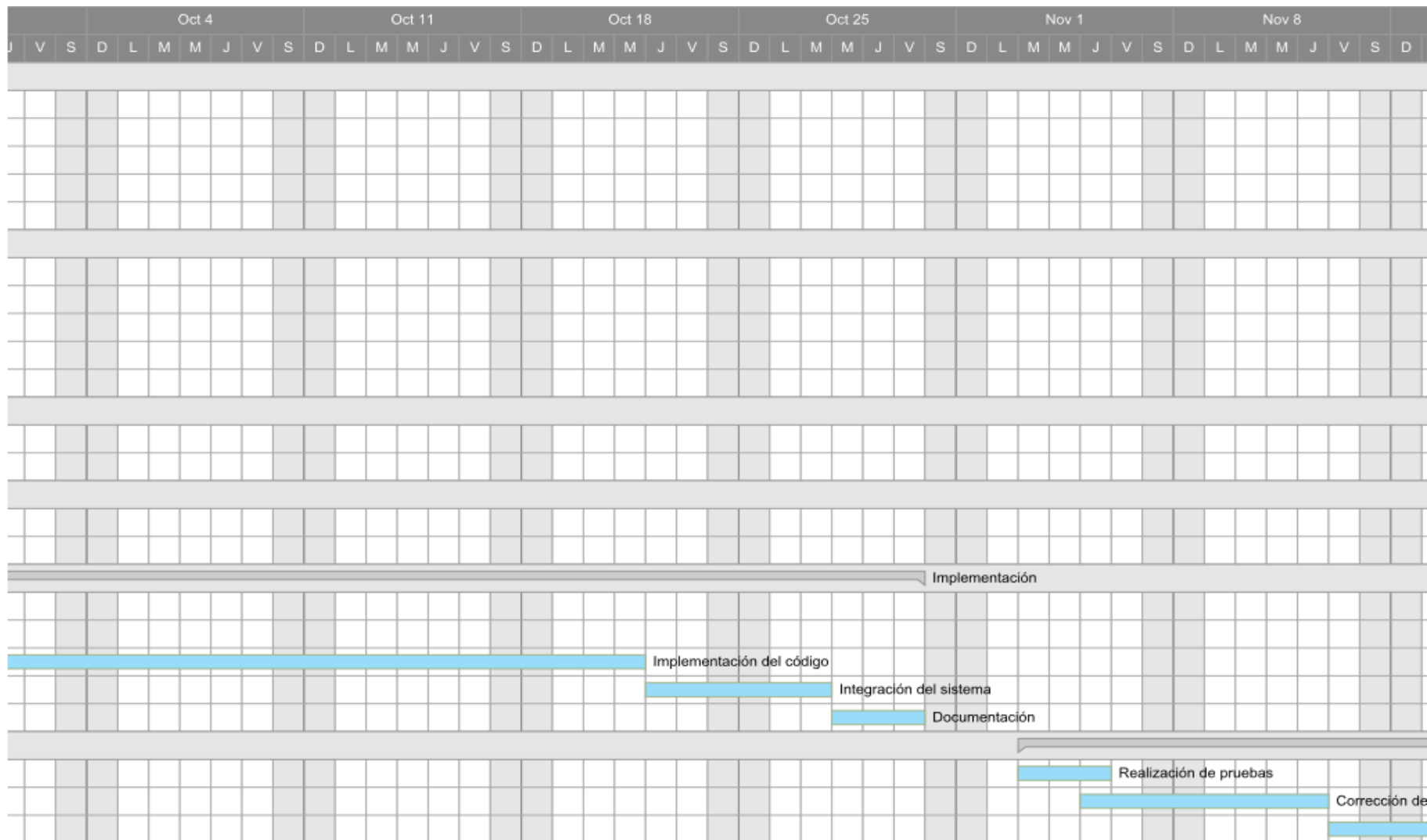


Ilustración 8 - Gantt 6

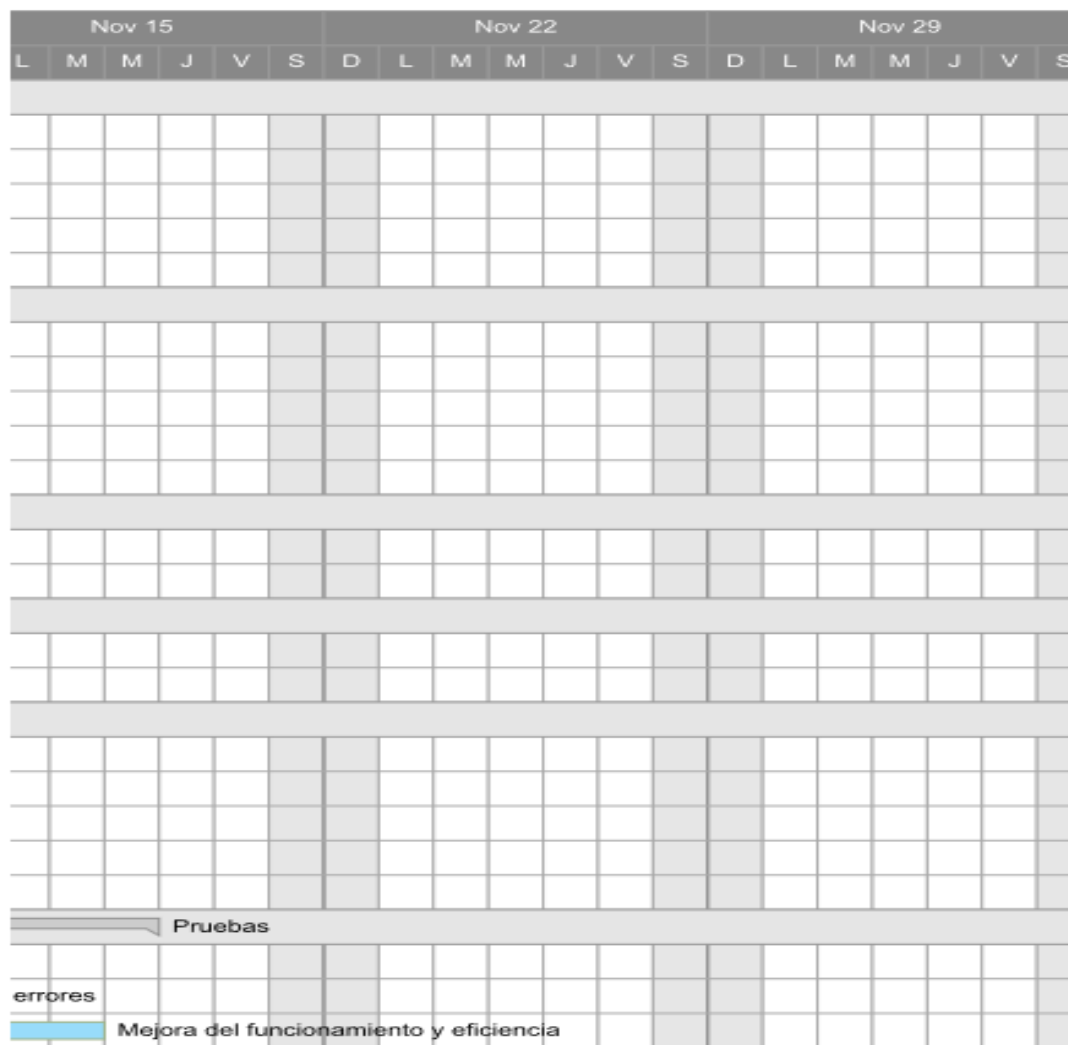


Ilustración 9 - Gantt 7

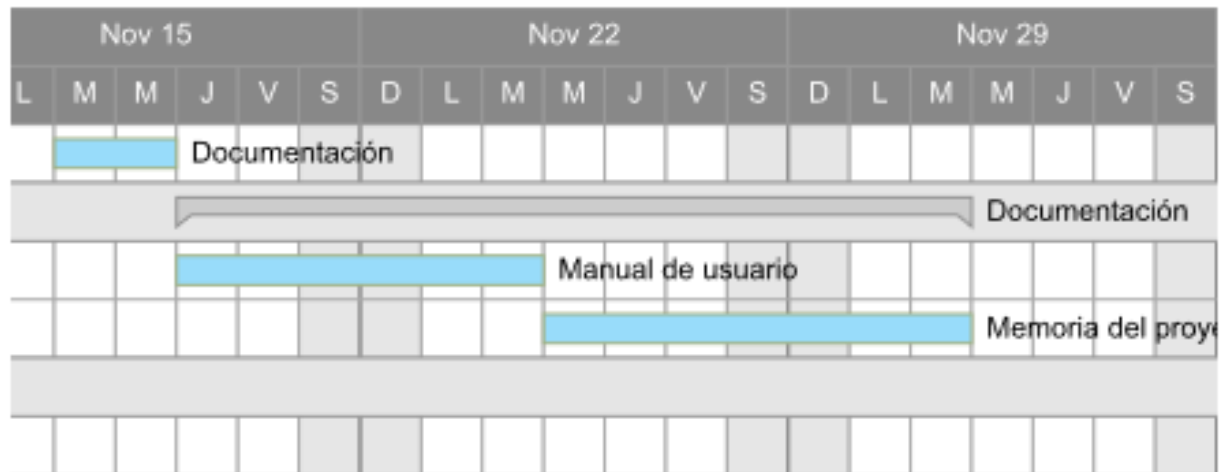


Ilustración 10 - Gantt 8

8. RIESGOS

En este apartado se identificarán los riesgos que pueden presentarse a lo largo de la realización del proyecto. Todo proyecto tiene sus riesgos y han de ser tenidos en cuenta, por tanto, conocerlos de antemano puede evitar que sucedan o tratar de minimizarlos al máximo.

Por ello se estudiará la probabilidad de que ocurra, el impacto que causaría y medidas correctoras o preventivas que se podrían llevar a cabo.

Los riesgos a tener en cuenta son los siguientes:

RIESGO	Retraso en la planificación temporal establecida al principio del proyecto
Probabilidad	Muy alta
Impacto	Retrasaría los plazos posteriores en la planificación temporal y en última instancia la fecha de finalización del proyecto
Prioridad	Alta
Prevención	Reuniones de seguimiento para asegurar que se cumplen los plazos preestablecidos
Plan de contingencia	Restablecer la planificación y valorar por qué se retrasa

Tabla 28 - Riesgo 1

RIESGO		Pérdida del contenido del proyecto desarrollado hasta el momento
Probabilidad	Muy baja	
Impacto	El impacto dependería de en qué fase del proyecto nos encontráramos si este riesgo llegara a hacerse efectivo. Si tuviera lugar al principio el impacto sería mínimo, pero de ocurrir en las fases finales del desarrollo, el impacto sería de gran importancia	
Prioridad	Alta	
Prevención	Creación periódica de copias de seguridad de datos	
Plan de contingencia	Restablecer la información haciendo uso de dichas copias de seguridad y valorar cómo tuvo lugar la pérdida de información para evitar que se produzca de nuevo en un futuro	

Tabla 29 - Riesgo 2

RIESGO		Problemas de salud o de tipo personal
Probabilidad	Media	
Impacto	Sería de mayor o menor gravedad en función de la duración de la enfermedad y de la eficiencia con la que lleváramos a cabo nuestro plan de contingencia	
Prioridad	Muy baja	
Prevención	Por lo general no podemos prevenir una enfermedad lo suficientemente importante como para que aparte a alguien de su trabajo	
Plan de contingencia	Realizar horas no definidas en la planificación temporal para recuperar el tiempo perdido	

Tabla 30 - Riesgo 3

RIESGO	
Pérdida o inutilidad de los instrumentos de trabajo como pueden ser la entrada de virus en el sistema operativo, descarga eléctrica en caso de tormenta, etc.	
Probabilidad	Muy baja
Impacto	Inutilidad del equipo, posible pérdida de información, etc.
Prioridad	Media
Prevención	Tener protegido el equipo con un buen antivirus
Plan de contingencia	Reinstalación del sistema operativo junto con los programas necesarios para el proyecto y restablecer copias de seguridad

Tabla 31 - Riesgo 4

RIESGO	
Estancamiento en el desarrollo de la aplicación debido al desconocimiento de la plataforma	
Probabilidad	Muy alta
Impacto	Pérdida de tiempo
Prioridad	Media
Prevención	Realizar una buena búsqueda de bibliografía en la que poder que nos guíen por el buen camino
Plan de contingencia	Recopilar buena información, si es posible consultar con expertos en la materia

Tabla 32 - Riesgo 5

9. EVALUACIÓN ECONÓMICA

A continuación, se detallará un estudio sobre la inversión económica que habrá que realizar en el proyecto, así como el planteamiento de distribución del sistema para la recuperación de la inversión y la obtención de beneficios.

Se recoge de antemano, que la empresa está formada por el alumno, por lo tanto, el único miembro asume todos los costes de la inversión y desempeña todas las funciones a realizar.

9.1 Inversión inicial

Para calcular el gasto inicial que nos va a suponer la realización del proyecto, nos centraremos en la revisión de los siguientes puntos:

- **Mano de obra:** es el gasto que se realiza durante el desarrollo de la aplicación.
 - Número de horas que se prevén invertir en el desarrollo -> **533 horas**
 - Precio por hora del programador -> Valoramos como está el mercado en la actualidad para un Ingeniero Informático y suponemos un salario bruto de 24.000€/año trabajando aproximadamente 1.800 horas, lo que supondría $24.000/1.800 = 13,33 \text{ €/hora}$.

Por lo tanto, para la realización de nuestro sistema valoramos la mano de obra en $533 \text{ horas} \times 13,33\text{€/hora} = 7.104,89 \text{ €}$.



- **Hardware:** son los dispositivos que utilizaremos durante el desarrollo.
 - Ordenador: a pesar de contar con nuestro ordenador personal antes de empezar para poder empezar, al hacer un uso exhaustivo del mismo, deberemos calcular el gasto de amortización. Considerando una vida útil de 4 años, precio de 750€, la duración proyecto será de

42 semanas y su porcentaje de uso será de un 100%. El gasto de amortización será $((750\text{€} / 4 \text{ años}) \times 42 \text{ semanas} \times 1 \text{ (porcentaje de uso)}) / 52 \text{ semanas del año} = 151,44 \text{ €}$.

- Lector DNI: dado que no disponemos de uno, necesitamos comprar uno nuevo, el valor del lector supone **10 €**.

El coste total del hardware es de **161,44 €**.



- **Software:** es el software de pago utilizado, en nuestro caso la mayoría se ha utilizado software libre.
 - Office 365 Estudiantes: 79€ durante 4 años. La amortización de esta licencia en nuestro proyecto es $(79\text{€} / 4 \text{ años}) \times 42 \text{ semanas} / 52 \text{ semanas del año} = 15,95 \text{ €}$.



- **Otros gastos:** son aquellos pequeños gastos que serán necesarios durante el desarrollo.
 - Luz: 40 €.
 - Copistería: 30 €.
 - Transportes: 20 €.

La estimación de estos gastos asciende a **90 €**.



COSTE TOTAL DEL PROYECTO

GASTOS	COSTE
Mano de obra	7.104,89
Hardware	161,44
Software	15,95
Otros gastos	90
TOTAL	7.372,28

Tabla 33 - Inversión proyecto

El coste total del proyecto es de **7.372,28 €**.

9.2 Recuperación de la inversión

Dado que nuestro desarrollo no es un sistema personalizado para una empresa, puede adaptarse a numerosos colegios que tengan las necesidades que el sistema cubre, si conseguimos poder vender a más de un cliente, se podría ajustar al máximo para conseguir un precio competitivo, darnos a conocer en el mundo de la educación y seguir creciendo y dando soporte a los clientes.

Pero como empezamos de cero, hay que ser realistas y no poner las expectativas muy altas, limitarnos a recuperar la inversión y obtener pocos beneficios, para que más adelante cuando estemos más integrados y podamos vender más, obtener mayor ganancia.

10. LEY DE PROTECCIÓN DE DATOS

10.1 Introducción

La Ley Orgánica 15/1999 de 13 de diciembre de Protección de Datos de Carácter Personal (LOPD), es la ley que garantiza y protege el tratamiento de los datos personales, las libertades públicas, los derechos de las personas físicas, su honor, intimidad y privacidad, personal y familiar. Teniendo como objetivo el tratamiento de los datos y ficheros personales de manera segura, velando por los derechos de los ciudadanos sobre ellos y las obligaciones de aquellos que los crean o tratan.



10.2 Aplicación de la Ley en nuestro Sistema

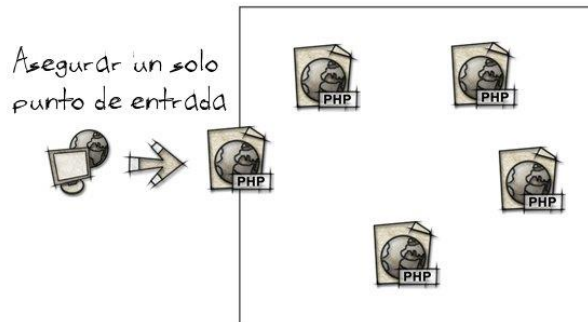
El principal motor de este proyecto es el tratamiento de datos personales, pero, sobre todo la mayoría de los datos que se manejan son datos pertenecientes a menores, por ellos hay que tener especial cuidado a la hora de proteger el sistema y hacerlo especialmente seguro, para que la ley se cumpla.

Para ello, se han optado por una serie de aspectos a la hora de la implementación del desarrollo:

- **Uso de sesiones:** Las sesiones en php permiten que se recuerden determinados datos al pasar de una página a otra, sin necesidad de pasarlos de manera oculta. Esto nos permite crear una sesión diferente cada vez que un usuario se identifique en el sistema, impidiendo acceder a páginas o datos a través del navegador sin una sesión iniciada, además estas tienen un límite de tiempo determinado que, sobrepasándolo, la sesión se acaba, siendo imposible que, si

una persona se deja una sesión abierta sin cerrar sesión al cabo de ese tiempo, pueda navegar por ella.

Aun así, se aconseja a los usuarios del sistema que, por seguridad, siempre que inicien sesión, sobre todo si no se encuentran en un dispositivo personal y privado, y no necesiten realizar ningún trámite, la cierren inmediatamente para evitar que una persona que venga detrás pueda consultar datos personales.



- **Protocolo HTTPS:** El protocolo de Transferencia de Hiper-Texto (HTTPS) es la versión segura del http (Hyper Text Transfer Protocol), permitiendo realizar transacciones de manera segura. La página web codifica la sesión a través de un certificado digital, dando al usuario una garantía de que la información que se envíe desde el sistema no va ser interceptada por terceros. Estos certificados de seguridad son conocidos como SSL (Secure Socket Layer Certificado), que, al instalarlos en nuestra página, en el navegador aparecerá un candado que mostrará la información del certificado que da credibilidad a nuestra web como segura.

En la barra de direcciones tendremos el inicio **https://**, impidiendo que introduciendo **http://** se acceda a nuestro sitio.



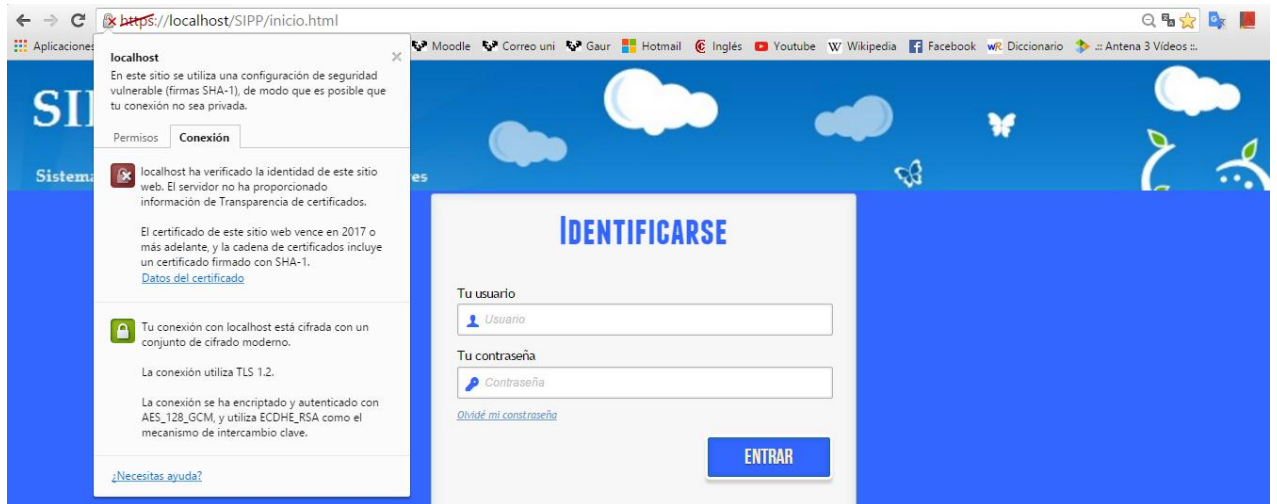
- **Identificación a través de DNI electrónico:** Desde un primer momento se planteó la necesidad de un sistema extra de seguridad en la aplicación, al principio esta característica se pensó utilizarla para realizar transacciones como firmar notas, circulares, etc. Pero tras replantearnos la finalidad de este proyecto, que era ante todo hacer una aplicación que protegiera los datos personales de los menores, se modificó para que no sólo se necesitase la firma electrónica para realizar gestiones, si no directamente que se utilizase para la identificación del usuario, permitiendo que todos los datos de los diferentes usuarios queden protegidos.



10.3 Dificultades de la implementación en nuestro Sistema

Una de las dificultades que nos encontramos a la hora de la implementación, fue en la configuración del protocolo HTTPS. Lo primero que se necesita para poder proteger la web es un certificado SSL, hoy en día la única opción que permite proteger con un certificado válido que identifique nuestra web con el candado verde, es pagando por él.

Sin embargo, xampp nos permite obtener un certificado SSL con contraseña con el que proteger nuestra web.



Como se puede observar, el navegador nos dice que a pesar de tener un certificado y estar cifrada la web, no puede validarlo como un sitio seguro porque no conoce la identidad de este. Sin embargo, si se trata de un sitio seguro, sólo pudiendo acceder a los datos si se conociese la clave del certificado.

Para dar más fiabilidad al sistema, si lo implementásemos en un centro de manera real, lo primero que tendríamos que hacer sería invertir en un certificado válido.

11. DNI-E

11.1 Introducción

Antes de comenzar el proyecto, la idea de hacer un sistema educativo estaba clara, a pesar de que sea un territorio en el que parece que ya está todo inventado, hay muchas lagunas en respecto a temas de seguridad. El fin era conseguir un sistema que permitiera a padres ser conscientes de todo lo relacionado con su hijo en el ámbito escolar, sin necesidad de tener a sus hijos de intermediarios, con agendas o papeles.

Tras realizar consultas con padres que tienen a sus hijos en distintos colegios, nos encontramos desde centros que no cuentan con ningún sistema, otro que tienen sistemas básicos en los que consultar las notas finales de sus hijos o incluso algunos que para dar a conocer notas a los padres las envía a través de correo electrónico.

Lo más preocupante que muchos padres no son conscientes que esto vulnera la ley de protección de datos, en muchos casos se juega con información confidencial de menores que puede llegar a manos de cualquier persona. Por ello, la solución más eficaz para asegurar que el destinatario de la información sea realmente la persona a la que va dirigido, se planteó la necesidad de usar el DNI electrónico para la identificación y firma de documentos de manera online.



El DNI es el Documento Nacional de Identidad, emitido por la Dirección General de la Policía, que acredita que la persona que lleva consigo este documento es quien dice

ser. A lo largo de los años el DNI ha ido evolucionando y adaptándose a las nuevas tecnologías, convirtiéndose hoy en día en un documento electrónico que permite acreditar a una persona no sólo físicamente. Algunas de las transacciones que se pueden realizar vía internet son:

- Hacer trámites completos en las Administraciones Públicas
- Realizar transacciones seguras en entidades bancarias
- Firmar documentos que luego van a ser enviados por correo electrónico

11.2 Necesidades

A pesar de no estar tan generalizado el uso del DNI electrónico, su uso es muy fácil, sólo se necesitan una serie de elementos:

- **Lector de tarjetas inteligente:** a través de conector USB, es el más común, aunque hoy en día también existen teclados que llevan lector de tarjetas integrado.



Sería recomendable asegurarse que sea compatible con el DNI electrónico, que verifique al menos:

- Cumpla el estándar ISO 7816 (1, 2 y 3).
- Soporta tarjetas asíncronas basadas en protocolos T=0 (y T=1).
- Soporta velocidades de comunicación mínimas de 9.600 bps.
- Soporta los estándares:
 - API PC/SC (Personal Computer/Smart Card)
 - CSP (Cryptographic Service Provider, Microsoft)

- API PKCS#11

- **Sistema operativo:**
 - Microsoft Windows (XP, Vista, 7 y 8)
 - Linux
 - Unix

- **Navegadores:**
 - Microsoft Internet Explorer (Versión 6.0 o superior)
 - Chrome
 - Mozilla Firefox (Versión 1.5 o superior)
 - Netscape (Versión 4.78 o superior)

- **Controladores / Módulos criptográficos:** Para que el lector sea leído correctamente, el navegador tiene que tener instalados unos elementos software determinados que pueden ser descargados en www.dnielectronico.es/descargas, aunque hoy en día la mayoría de los lectores automáticamente son capaces de realizar la lectura utilizando sus propios drivers, sin necesidad de estos módulos. En las instrucciones del lector vendrá explicado detalladamente.

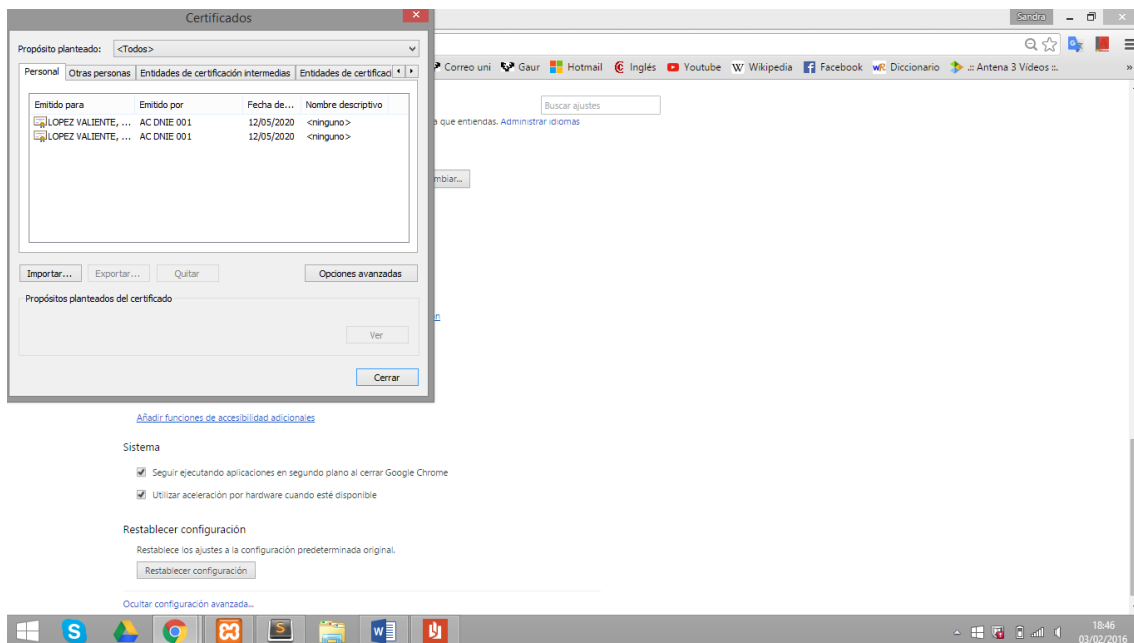
11.3 Uso del DNI en nuestro Sistema

Primero antes de nada tenemos que tener claro que la firma electrónica certificada, es aquella que ha sido expedida por una Autoridad Certificadora que permite asegurar la integridad del titular.

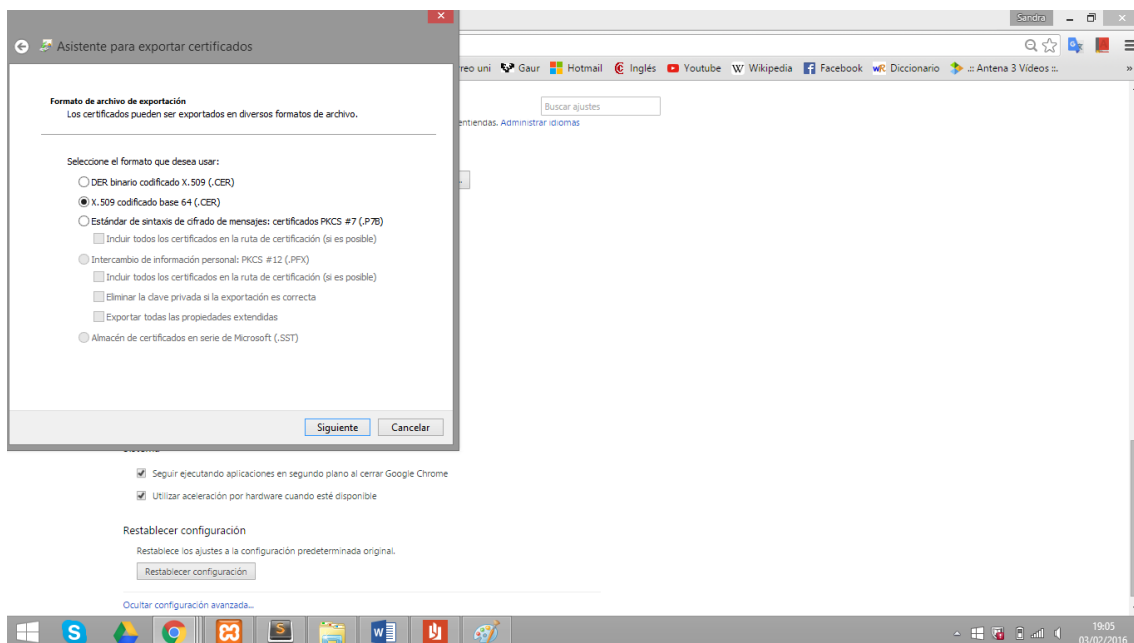
Para poder utilizar dicha firma, necesitamos el certificado de la misma. Existen varias maneras de obtener el certificado electrónico de nuestro DNI, por ejemplo, solicitarlo en la administración pública para que te lo expidan, sin embargo, gracias al DNI electrónico y un lector de tarjetas, seremos capaces de obtener el certificado que necesitamos para realizar la firma y dar autenticidad a nuestro sistema.

Una vez hayamos seguido las instrucciones del fabricante del lector y esté configurado para comenzar a hacer uso de él, introduciremos en el lector de tarjetas

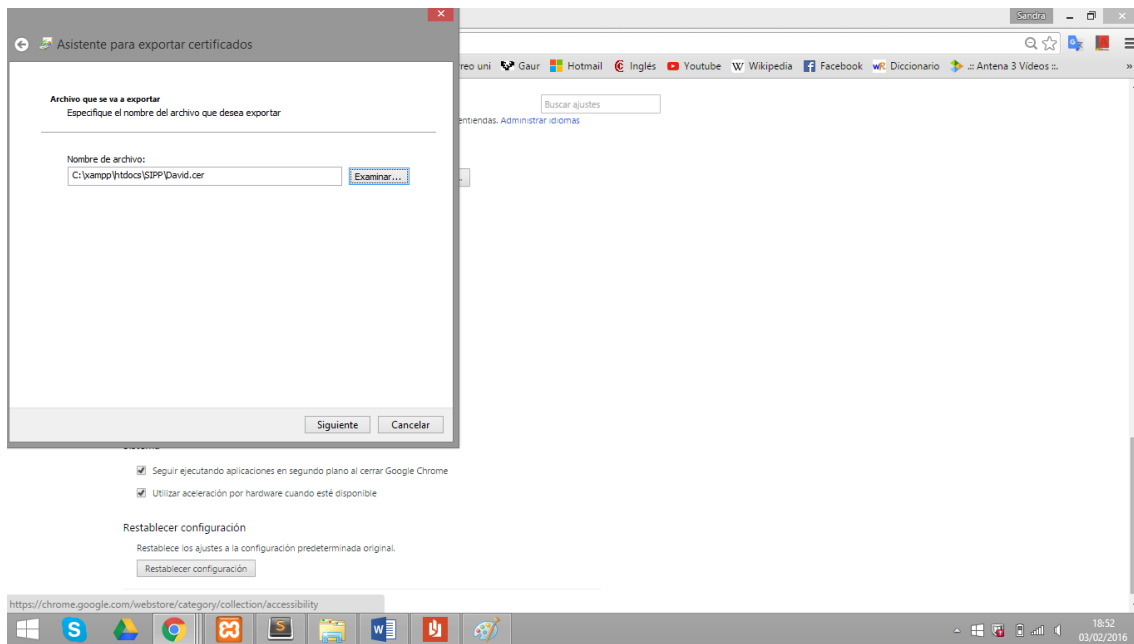
nuestro DNI. Iremos al navegador e iremos en configuración a la pestaña de certificados, como podemos observar el lector nos lee automáticamente el certificado:



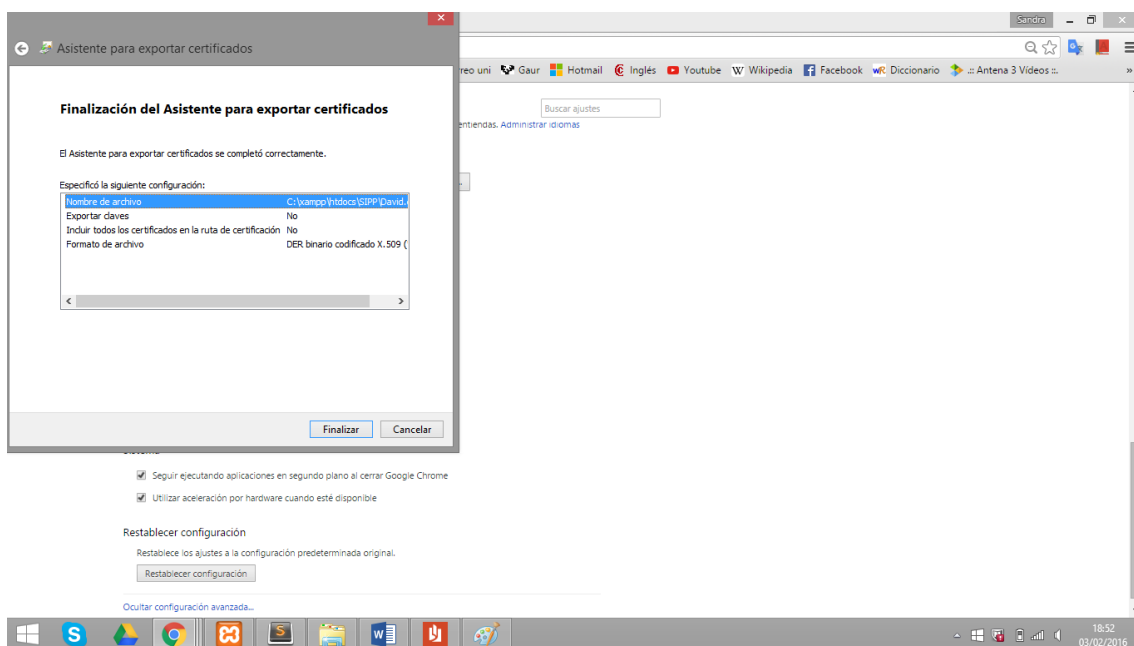
Para poder utilizarlo en nuestra aplicación, necesitaremos exportar el certificado con extensión .cer, para ello lo seleccionaremos y pulsamos exportar



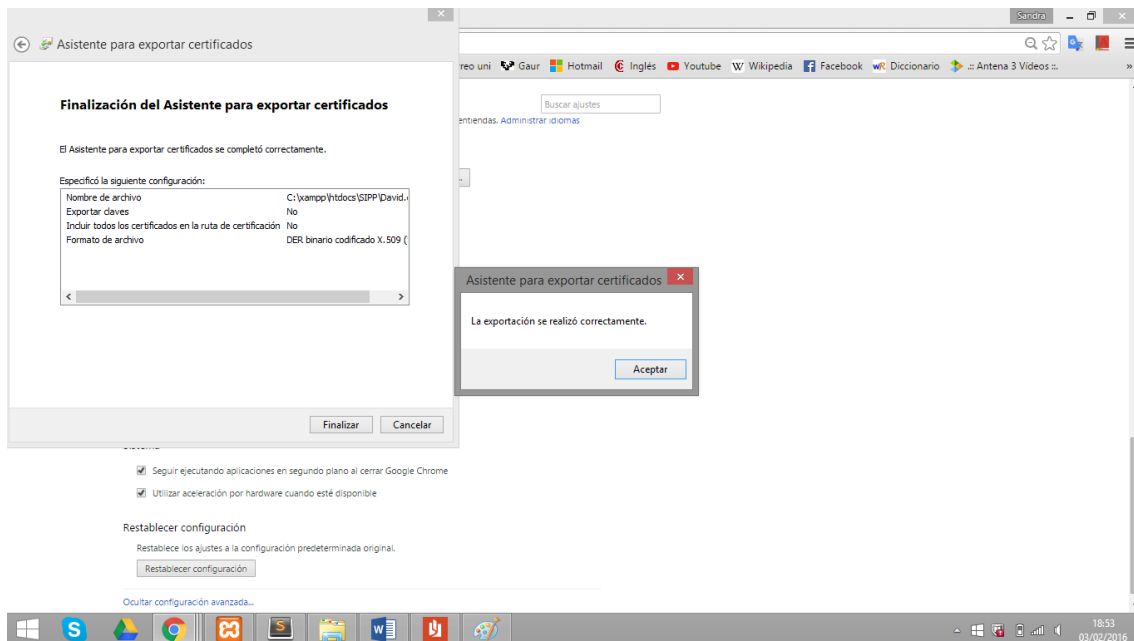
Seleccionamos el tipo de extensión y siguiente



Seleccionamos una carpeta donde almacenarlo, asignarle un nombre y pulsar siguiente



Finalizar



Ahora tendremos ya el certificado listo para poder usarlo en nuestro sistema.

11.4 Dificultades de la implementación en nuestro Sistema

A pesar de que el DNI electrónico se implantó ya hace bastantes años, no es una técnica que esté muy extendida y la información que existe no está muy completa. Por ello, se ha necesitado mucha investigación acerca del tema, no sólo en como incorporar esta funcionalidad en el sistema si no como es el funcionamiento del DNI electrónico, la firma y sus certificados.

Además, uno de los problemas que más quebraderos de cabeza nos ha dado, fue la lectura del certificado, ya que este es un archivo que cada usuario va a tener que obtener a través de su DNI electrónico, por lo tanto, el sistema tiene que hacer esta lectura del fichero y por seguridad los navegadores no permiten obtener las rutas locales de los dispositivos. Por ello se planteará la necesidad de tener una carpeta raíz que sea común en cualquier dispositivo donde poder alojar nuestro certificado para seleccionarlo en la ruta de archivos desde el sistema para conocer el nombre del mismo.

12. UNIFICACIÓN DEL DESARROLLO

12.1 Captura de requisitos

En este apartado se enumeran los requisitos funcionales que debe tener la aplicación respecto a las necesidades de los diferentes actores que componen el sistema.

12.1.1 Requisitos del Administrador

- Gestionar alumnos: Añadir, Editar y eliminar.
- Gestionar usuarios: Añadir, Editar y eliminar.
- Gestionar profesores: Añadir, Editar y eliminar.
- Gestionar cursos: Añadir, Editar y eliminar.
- Gestionar asignaturas: Añadir, Editar y eliminar.
- Gestionar circulares: Añadir, Editar y eliminar.

12.1.2 Requisitos del Profesor

- Gestionar alumnos:
 - Ver información de los alumnos que están matriculados en las asignaturas que imparte.
 - Consultar las notas de las tareas y exámenes de los alumnos de las asignaturas que imparte, así como establecer una nota final para los diferentes trimestres.
 - Añadir y eliminar faltas a los alumnos en las diferentes asignaturas que imparte, así como ver las justificaciones de dichas faltas.
- Mensajes:
 - Enviar, recibir y responder mensajes a los diferentes alumnos de las asignaturas que imparte.
- Gestionar deberes:
 - Añadir, editar y eliminar deberes, para las diferentes asignaturas que imparte.

- Consultar deberes pendientes que no han sobrepasado el límite de entrega.
- Corregir los deberes existentes para los diferentes alumnos de las asignaturas que imparte.
- Gestionar exámenes:
 - Añadir, editar y eliminar exámenes, para las diferentes asignaturas que imparte.
 - Consultar exámenes pendientes que no han sobrepasado el límite de entrega.
 - Corregir los deberes existentes para los diferentes alumnos de las asignaturas que imparte.
- Gestionar tutoría (sólo si el profesor es tutor de algún curso):
 - Enviar las circulares existentes en el sistema a los alumnos de los que es tutor.
 - Añadir fechas y horas disponibles para que los padres de los alumnos de los que es tutor puedan reservar cita y ver las citas que han sido reservadas y el motivo de estas.
 - Consultar las notas finales de cada trimestre de las asignaturas de los alumnos de los que es tutor para prepararlas y enviarlas a los padres para que procedan a su firma.

12.1.3 Requisitos del Padre

- Gestionar hijos:
 - Ver y editar la información básica de todos sus hijos.
 - Consultar las notas finales de cada trimestre de todos sus hijos, así como poder firmarlas.
 - Ver las circulares enviadas por el tutor de todos sus hijos, así como poder firmarlas.
 - Consultar las horas disponibles para cita del tutor de sus hijos y poder reservarlas.
- Gestionar curso escolar:

- Ver los deberes pendientes de las asignaturas que tienen sus hijos.
- Ver las notas de los deberes corregidos por el profesor de las asignaturas de sus hijos.
- Ver los exámenes pendientes de las asignaturas que tienen sus hijos.
- Ver las notas de los exámenes corregidos por el profesor de las asignaturas de sus hijos.
- Ver las faltas de sus hijos en las diferentes asignaturas, así como poder justificarlas.
- Mensajes:
 - Enviar, recibir y responder mensajes a los diferentes alumnos de las asignaturas que imparte.

12.2 Casos de uso

A continuación, se muestra el diagrama de casos de uso de los diferentes actores que componen nuestro sistema. Más adelante se detallan los diferentes casos de uso extendidos.

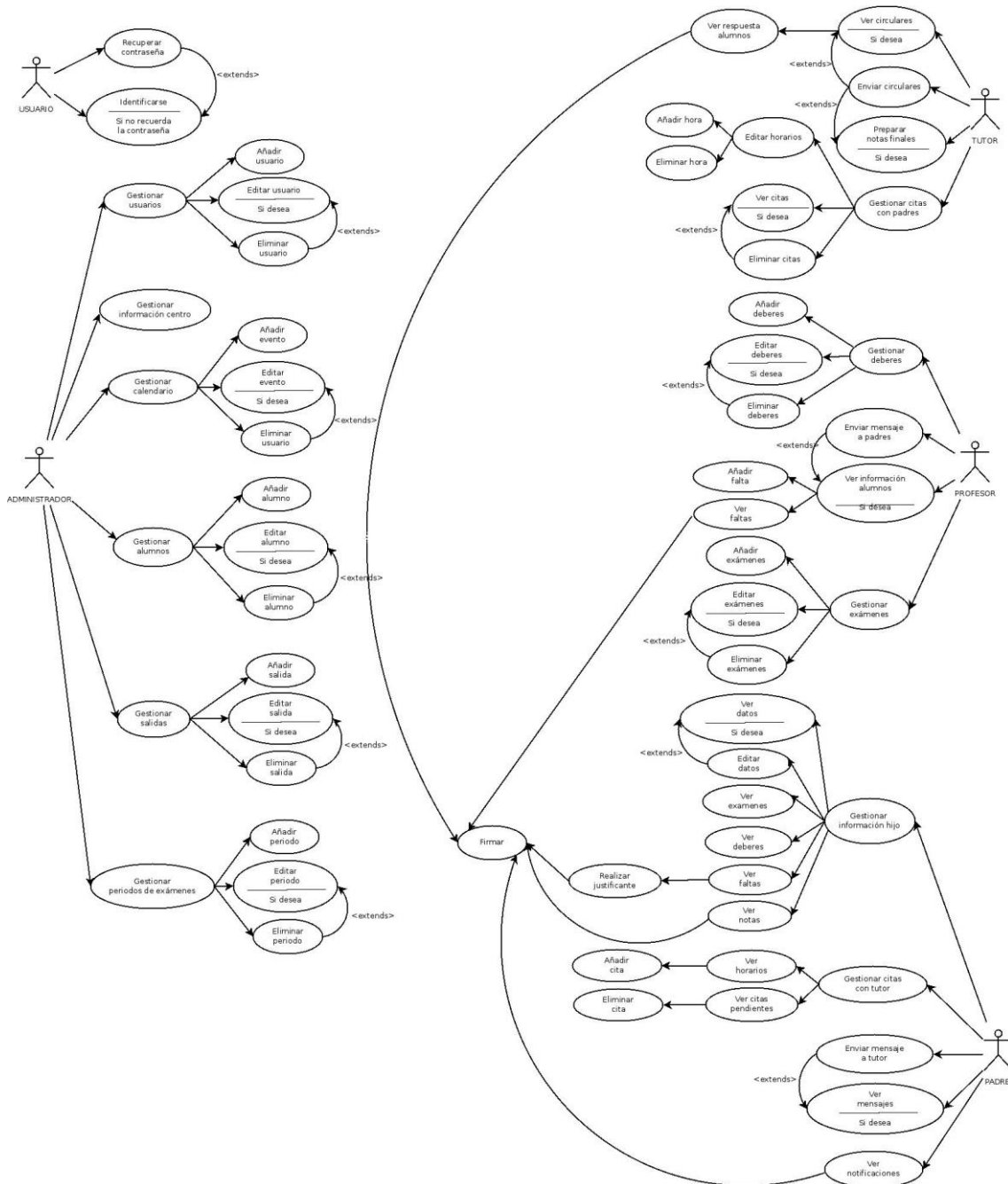


Ilustración 11 - Casos de uso

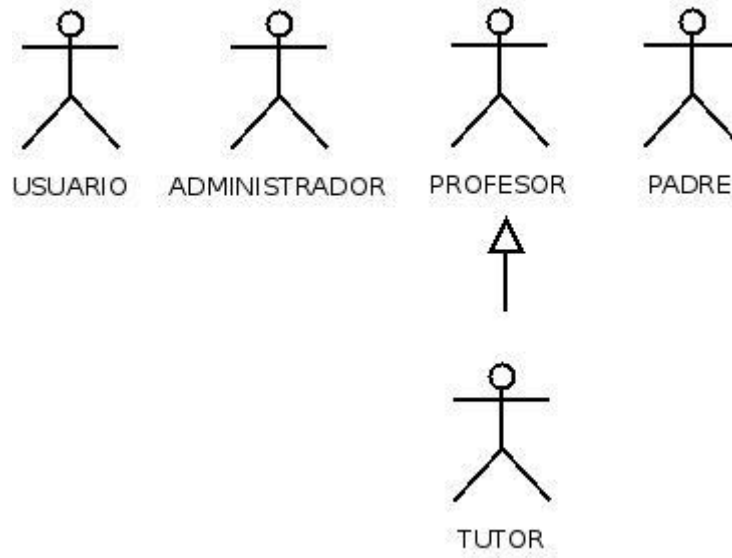


Ilustración 12 - Jerarquía actores

12.2.1 Casos de uso del administrador



Nombre:	Añadir alumnos
Descripción:	El administrador podrá añadir alumnos
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar alumnos" 2. El administrador pulsa "Añadir alumno" 3. El administrador rellena los datos del alumno y pulsa "Añadir" <p>[Si los campos introducidos no tienen el formato adecuado]</p> <ol style="list-style-type: none"> 3a. Aparecerá un mensaje de error <p>[Si el alumno introducido ya existe]</p> <ol style="list-style-type: none"> 3b. Aparecerá un mensaje de error <p>[Si todo es correcto]</p> <ol style="list-style-type: none"> 3c. Se añadirá un nuevo alumno
Postcondiciones:	Se añadirá un nuevo alumno a nuestro sistema

Tabla 34 - Caso de uso Administrador - Añadir Alumno



Nombre:	Editar alumnos
Descripción:	El administrador podrá editar alumnos
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar alumnos" 2. El administrador pulsa "Editar alumno" 3. El administrador selecciona el usuario a editar 4. Rellena los datos del alumno a modificar y pulsa "Editar" [Si los campos introducidos no tienen el formato adecuado] <ol style="list-style-type: none"> 4a. Aparecerá un mensaje de error [Si todo es correcto] 4b. Se modificará el alumno
Postcondiciones:	Se editará la información del alumno en nuestro sistema

Tabla 35 - Caso de uso Administrador - Editar Alumno



Nombre:	Eliminar alumnos
Descripción:	El administrador podrá eliminar alumnos
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar alumnos" 2. El administrador pulsa "Eliminar alumno" 3. El administrador selecciona el usuario a eliminar <ol style="list-style-type: none"> 3b. Se le mostrará un mensaje de confirmación de eliminación, si pulsa "Aceptar" se eliminará
Postcondiciones:	Se eliminará al alumno de nuestro sistema

Tabla 36 - Caso de uso Administrado - Eliminar Alumno



Nombre:	Añadir salida
Descripción:	El administrador podrá añadir salidas de los alumnos
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar salidas" 2. Se carga una lista con las salidas 3. Si el administrador rellena los datos y pulsa "Añadir" [Si la salida introducida ya existe] 3a. Aparecerá un mensaje de error [Si todo es correcto] 3b. Se añade una salida a la lista
Postcondiciones:	Se añadirá una nueva salida a nuestro sistema

Tabla 37 - Caso de uso Administrador - Añadir salida00



Nombre:	Editar salidas
Descripción:	El administrador podrá editar las salidas de los alumnos
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar salidas" 2. Se carga una lista con las salidas 3. Si el administrador selecciona una salida en la lista y pulsa "Editar" 3a. Modifica los campos necesarios y pulsa "Editar"
Postcondiciones:	Se editará la salida en nuestro sistema

Tabla 38 - Caso de uso Administrador - Editar salida



Nombre:	Eliminar salidas
Descripción:	El administrador eliminar las salidas de los alumnos
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar salidas" 2. Se carga una lista con las salidas 3. Si el administrador selecciona una salida y pulsa "Eliminar" 4a. Aparecerá un mensaje de confirmación [Si pulsa "Aceptar"] 4aa. Se eliminará la salida
Postcondiciones:	Se eliminará la salida de nuestro sistema

Tabla 39 - Caso de uso Administrador - Eliminar salida



Nombre:	Añadir usuarios
Descripción:	El administrador podrá añadir usuarios, como profesores, tutores o "padres"
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar usuarios" 2. El administrador pulsa "Añadir usuario" 3. El administrador elige el tipo de usuario entre profesor, tutor y alumno 3a. Se rellenan los datos que pide en función del tipo y se pulsa "Añadir"
Postcondiciones:	Se añadirá un nuevo usuario a nuestro sistema

Tabla 40 - Caso de uso Administrador - Añadir usuario



Nombre:	Editar usuario
Descripción:	El administrador podrá editar usuarios, como profesores, tutores o padres
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar usuarios" 2. El administrador pulsa "Añadir usuario" 3. El administrador elige el tipo de usuario entre profesor, tutor y alumno <ol style="list-style-type: none"> 3a. Se le cargará los usuarios que corresponden a ese tipo 3b. Se eligen los datos nuevos que queremos cambiar y se pulsa "Editar"
Postcondiciones:	Se editará la información del usuario en nuestro sistema

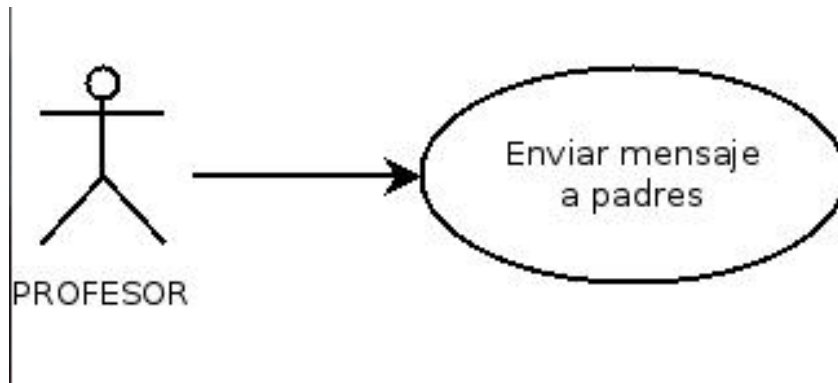
Tabla 41 - Caso de uso Administrador - Editar usuario



Nombre:	Eliminar usuario
Descripción:	El administrador podrá editar usuarios, como profesores, tutores o padres
Actores:	Administrador
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El administrador pulsa "Gestionar usuarios" 2. El administrador pulsa "Eliminar usuario" 3. El administrador elige el tipo de usuario entre profesor, tutor y alumno <ol style="list-style-type: none"> 3a. Se le cargará los usuarios que corresponden a ese tipo, selecciona y pulsa "Eliminar" 3b. Se le mostrará un mensaje de confirmación de eliminación, si pulsa "Aceptar" se eliminará
Postcondiciones:	Se eliminará al usuario de nuestro sistema

Tabla 42 - Caso de uso Administrador - Eliminar usuario

12.2.2 Casos de uso del profesor



Nombre:	Enviar mensaje a padres
Descripción:	El profesor podrá enviar a los padres de los alumnos a los que da clases mensajes
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Enviar mensaje" 2. El profesor elige el curso que desee y el alumno del curso elegido 3. El profesor escribe un mensaje y pulsa "Enviar"
Postcondiciones:	Se enviará el mensaje a los padres

Tabla 43 - Caso de uso Profesor - Enviar mensaje



Nombre:	Añadir deberes
Descripción:	El profesor podrá añadir deberes de las asignaturas de cada curso que da
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Gestionar deberes" 2. Se le cargará un listado con los deberes pendientes [Si el profesor rellena los datos y pulsa "Añadir deberes"] 2a. Se crearán unos nuevos deberes
Postcondiciones:	Se añadirá unos nuevos deberes a nuestro sistema

Tabla 44 - Caso de uso Profesor - Añadir deber



Nombre:	Editar deberes
Descripción:	El profesor podrá editar los deberes de las asignaturas de cada curso que da
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Gestionar deberes" 2. Se le cargará un listado con los deberes pendientes [Si selecciona unos deberes y pulsa "Editar deberes"] 2a. Rellenara con los nuevos datos y pulsa "Editar deberes"
Postcondiciones:	Se modificarán los deberes en nuestro sistema

Tabla 45 – Caso de uso Profesor – Editar deber



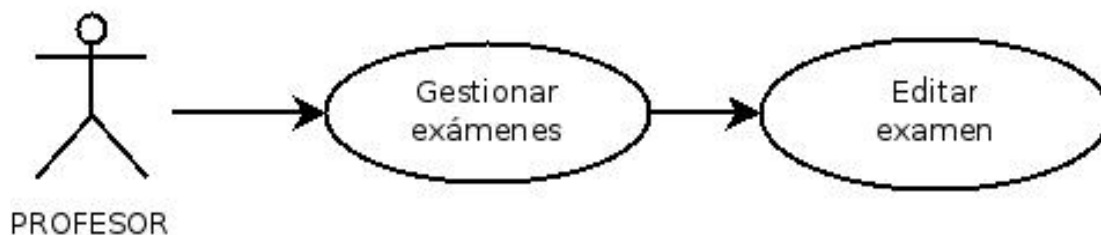
Nombre:	Eliminar deberes
Descripción:	El profesor podrá eliminar los deberes de las asignaturas de cada curso que da
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Gestionar deberes" 2. Se le cargará un listado con los deberes pendientes [Si selecciona unos deberes y pulsa "Eliminar deberes"] 2a. Se mostrará un mensaje de error, si pulsa "Aceptar" se eliminarán los deberes
Postcondiciones:	Se eliminarán los deberes en nuestro sistema

Tabla 46 - Caso de uso Profesor - Eliminar Deber



Nombre:	Añadir examen
Descripción:	El profesor podrá añadir exámenes de las asignaturas de cada curso que da
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Gestionar exámenes" 2. [Si el profesor rellena los datos y pulsa "Añadir examen"] <ol style="list-style-type: none"> [Si el examen introducido ya existe] <ol style="list-style-type: none"> 2a. Aparecerá un mensaje de error [Si todo es correcto] <ol style="list-style-type: none"> 2b. Se crearán un nuevo examen
Postcondiciones:	Se añadirá un nuevo examen a nuestro sistema

Tabla 47 - Caso de uso Profesor - Añadir examen



Nombre:	Editar examen
Descripción:	El profesor podrá editar los exámenes de las asignaturas de cada curso que da
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Gestionar exámenes" 2. [Si selecciona un examen y pulsa "Editar examen"] <ol style="list-style-type: none"> 2a. Rellenara con los nuevos datos y pulsa "Editar examen" <ol style="list-style-type: none"> [Si existe otro examen con las mismas características] <ol style="list-style-type: none"> 2a1. Aparecerá un mensaje de error [Si todo es correcto] <ol style="list-style-type: none"> 2a2. Se Modificará el examen
Postcondiciones:	Se modificará los datos del examen en nuestro sistema

Tabla 48 - Caso de uso Profesor - Editar examen



Nombre:	Eliminar examen
Descripción:	El profesor podrá eliminar los exámenes de las asignaturas de cada curso que da
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Gestionar exámenes" 2. [Si selecciona un examen y pulsa "Eliminar examen"] <ol style="list-style-type: none"> 2a. Se mostrará un mensaje de error, si pulsa "Aceptar" se eliminará el examen
Postcondiciones:	Ninguna

Tabla 49 - Caso de uso Profesor - Eliminar examen



Nombre:	Añadir falta
Descripción:	El profesor podrá añadir las faltas de ese alumno a sus asignaturas
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Mis alumnos" 2. El profesor elige el curso que desee y el alumno del curso elegido 3. Se cargará la información del alumno elegido 4. [Si el profesor pulsa "Añadir falta"] <ol style="list-style-type: none"> 4a. El profesor rellena la asignatura y la fecha y pulsa "Añadir falta"
Postcondiciones:	Se añadirá una nueva falta al alumno en el sistema

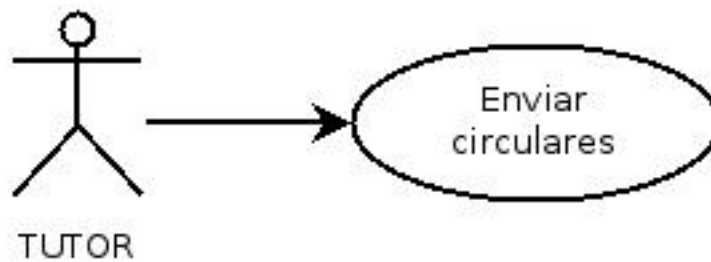
Tabla 50 - Caso de uso Profesor - Añadir falta



Nombre: Ver faltas	
Descripción:	El profesor podrá ver las faltas de ese alumno a sus asignaturas
Actores:	Profesor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El profesor pulsa "Mis alumnos" 2. El profesor elige el curso que desee y el alumno del curso elegido 3. Se cargará la información del alumno elegido 4. [Si el profesor pulsa "ver faltas"] <ol style="list-style-type: none"> 4a. Se cargará una lista con las faltas del alumno en las asignaturas del profesor, la fecha y si ha sido justificadas 4b. Puede añadir una falta, rellenando la asignatura y la fecha y pulsando "Añadir falta"
Postcondiciones:	Ninguna

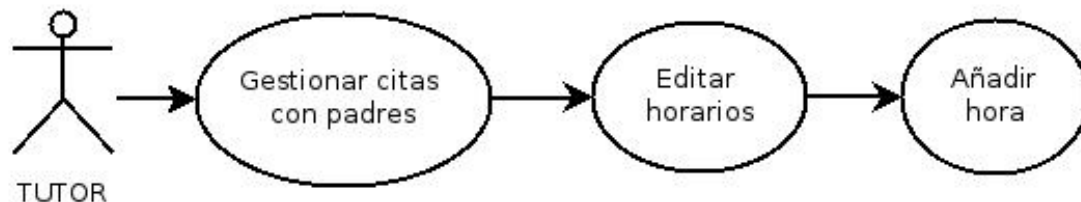
Tabla 51 - Caso de uso Profesor - Ver faltas

12.2.3 Casos de uso del tutor



Nombre:	Enviar circular
Descripción:	El tutor podrá enviar circulares a los padres, como excursiones, notificaciones importantes, notas finales...
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Enviar circular" 2. El tutor elige el tipo de circular que desee y la circular que quiere mandar y pulsa "Enviar circular"
Postcondiciones:	Se enviará una circular a todos los padres de sus alumnos

Tabla 52 - Caso de uso Tutor - Enviar Circular



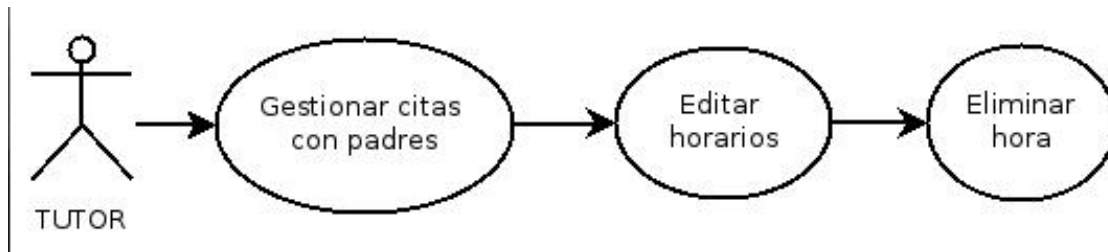
Nombre:	Añadir hora para citas
Descripción:	El tutor podrá añadir horarios para citarse con los padres
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Gestionar citas" 2. Se cargarán una lista con las citas concertadas por los padres y otra con los horarios establecidos [Si añade un nuevo horario y pulsa "Crear"] [Si los campos introducidos no tienen el formato adecuado] <ol style="list-style-type: none"> 2a. Aparecerá un mensaje de error [Si todo es correcto] 2b. Se añadirá un nuevo horario a la lista
Postcondiciones:	Se añadirá un nuevo horario al sistema

Tabla 53 - Caso de uso Tutor - Añadir hora



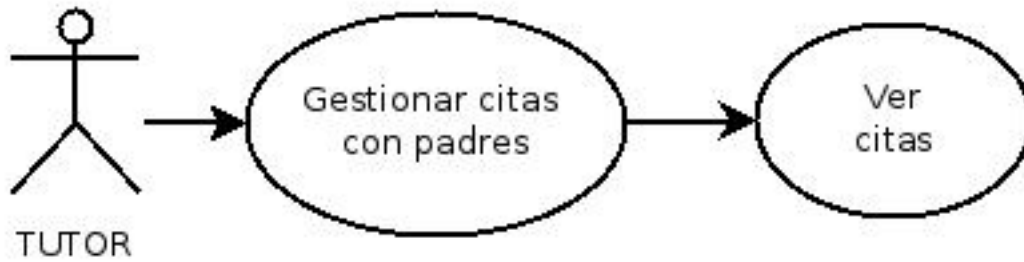
Nombre:	Eliminar citas con padres
Descripción:	El tutor podrá eliminar las citas que los padres han concertado con ellos
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Gestionar citas" 2. Se cargarán una lista con las citas concertadas por los padres y otra con los horarios establecidos [Si selecciona una cita concertada y pulsa "Eliminar cita"] <ol style="list-style-type: none"> 2a. Se mostrará un mensaje de confirmación y si pulsa "Aceptar" se eliminará la cita
Postcondiciones:	Se eliminará la cita en nuestro sistema y se le enviará un mensaje a los padres

Tabla 54 - Caso de uso Tutor - Eliminar cita



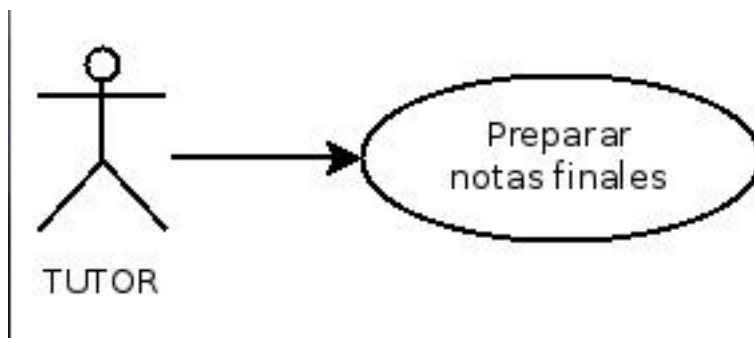
Nombre:	Eliminar hora para citas
Descripción:	El tutor podrá eliminar los horarios que tiene establecidos para citarse con los padres
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Gestionar citas" 2. Se cargarán una lista con las citas concertadas por los padres y otra con los horarios establecidos [Si selecciona un horario y pulsa "Eliminar hora"] <ol style="list-style-type: none"> 2a. Se mostrará un mensaje de confirmación y si pulsa "Aceptar" se eliminará el horario
Postcondiciones:	Se eliminarán los horarios del sistema

Tabla 55 - Caso de uso Tutor - Eliminar hora



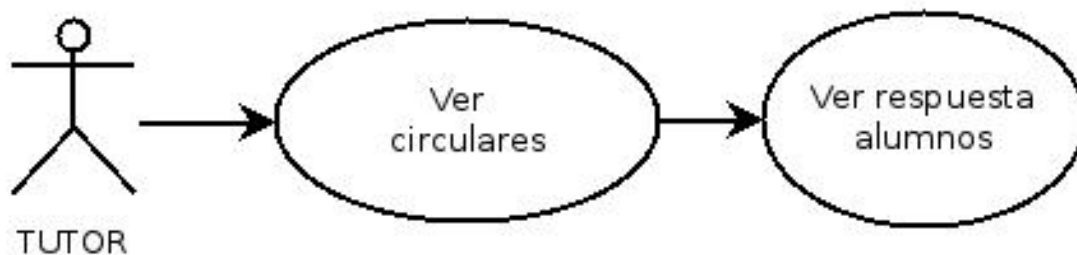
Nombre:	Ver citas con padres
Descripción:	El tutor podrá ver las citas que tiene concertadas con los padres
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Gestionar citas" 2. Se cargarán una lista con las citas concertadas por los padres y otra con los horarios establecidos
Postcondiciones:	Ninguna

Tabla 56 - Caso de uso Tutor - Ver citas



Nombre: Preparar notas finales	
Descripción:	El tutor podrá crear las notas finales de sus alumnos, con las notas puestas por el resto de profesores y añadiendo las observaciones personales
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Preparar notas finales" 2. El tutor elige el alumno del que preparar las notas 3. Se cargará un cuadro con las notas de sus asignaturas y las observaciones que cada profesor le ha dado al alumno seleccionado, el número de faltas, y las notas medias 4. El tutor rellenará los comentarios finales que dar al alumno <ul style="list-style-type: none"> [Si pulsa "Enviar"] <ul style="list-style-type: none"> 4a. Se enviará la circular a los padres [Si pulsa "Guardar"] <ul style="list-style-type: none"> 4b. Se guardará los datos introducidos
Postcondiciones:	Se guardarán en el sistema las notas y se enviará una circular a los padres

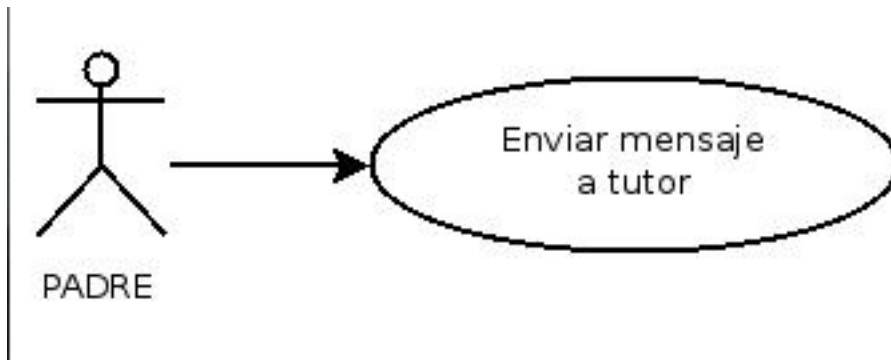
Tabla 57 - Caso de uso Tutor - Preparar notas



Nombre: Ver respuesta a circulares	
Descripción:	El tutor podrá ver y ver respuestas de las circulares, como excursiones, notificaciones importantes, notas finales...
Actores:	Tutor
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El tutor pulsa "Ver circulares" 2. El tutor elige el tipo de circular que desee y la circular que quiera 3. Se mostrará la información de la circular [Si la circular ya ha sido enviada y pulsa "Ver respuestas"] <ol style="list-style-type: none"> 3b. Se mostrará un listado con todos sus alumnos y la respuesta a la circular de los padres
Postcondiciones:	Ninguna

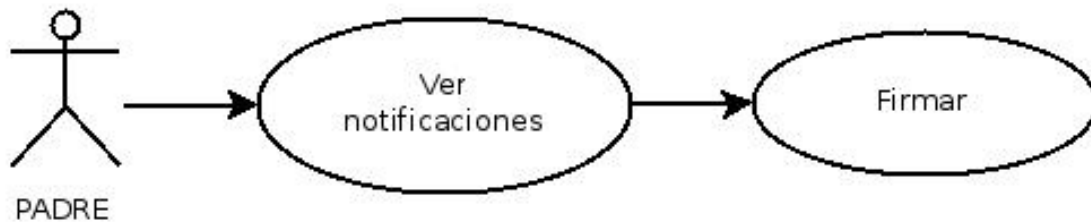
Tabla 58 - Caso de uso tutor - Ver respuesta

12.2.4 Casos de uso del padre



Nombre:	Enviar mensaje a tutor
Descripción:	El padre podrá enviar un mensaje al tutor de su hijo
Actores:	Padre
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Enviar mensaje" 2. El padre escribe un mensaje y pulsa "Enviar"
Postcondiciones:	Se enviará el mensaje al tutor

Tabla 59 - Caso de uso Padre - Enviar mensaje



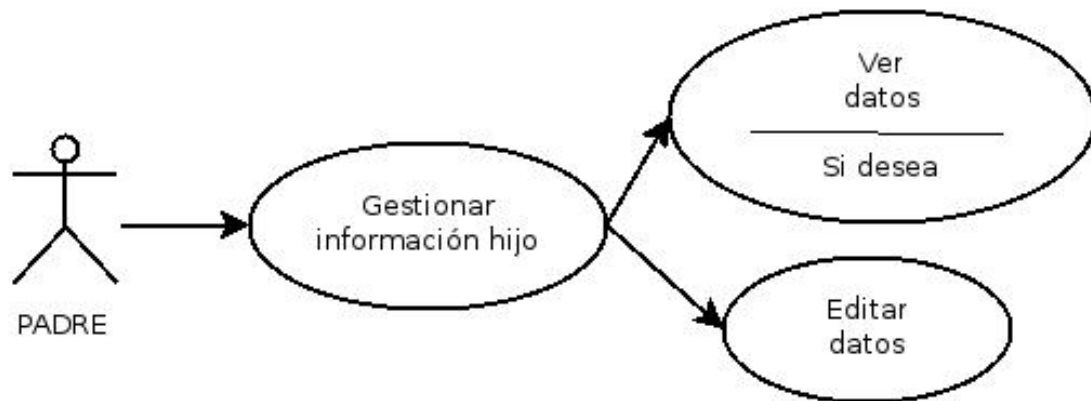
Nombre:	Firmar notificaciones
Descripción:	El padre podrá ver las circulares enviadas por el tutor, tendrá que firmarlas y en el caso de que sea una salida decir si su hijo asistirá o no.
Actores:	Padre
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Ver mensajes" 2. Se cargará una lista con todas las notificaciones que tiene pendientes de ver o firmar [Si selecciona una y pulsa "Ver notificación"] <ol style="list-style-type: none"> 2a. Le aparecerá el contenido [Si selecciona una y pulsa "Firmar"] [Si la circular es una excursión] <ol style="list-style-type: none"> 2a1. El padre tendrá que decir si asistirá o no 2b. Le aparecerá un mensaje de confirmación y si pulsa "Aceptar" la circular habrá sido firmada
Postcondiciones:	La circular quedará firmada en nuestro sistema

Tabla 60 - Caso de uso Padre – Firmar circulares



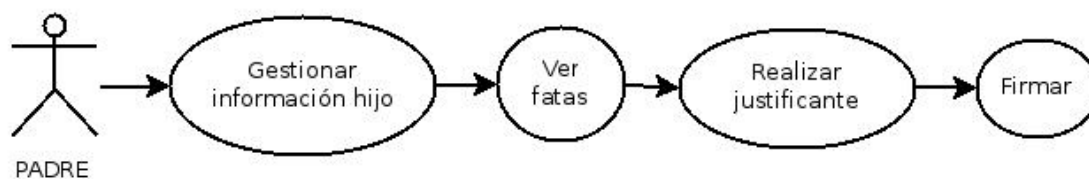
Nombre:		Firmar notas hijo
Descripción:	El padre podrá ver las notas de su hijo	
Actores:	Padre	
Precondiciones:	Ninguna	
Requisitos no funcionales:	Ninguno	
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Mi hijo" 2. Se cargará un menú [Si pulsa "Ver notas"] <ol style="list-style-type: none"> 2a. Primero seleccionará el trimestre del que quiere ver las notas finales 2b. Se le cargará una pantalla con las notas finales del alumno 2c. Revisa las notas y pulsa "Firmar" <ol style="list-style-type: none"> 2c1. Le aparecerá un mensaje de confirmación y pulsamos "Aceptar" para finalizar la firma 	
Postcondiciones:	Quedarán firmadas las notas en el sistema	

Tabla 61 - Caso de uso Padre - Firmar notas



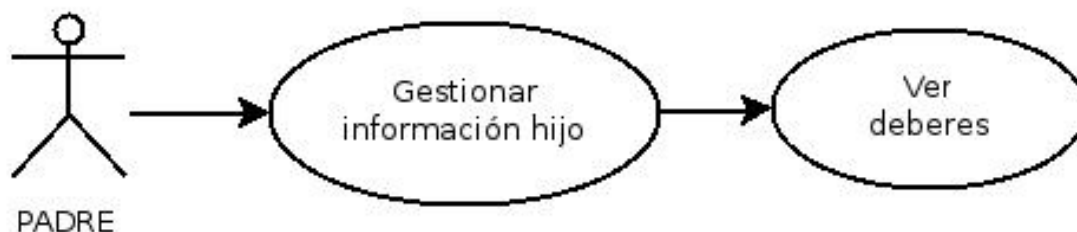
Nombre:		Ver información hijo
Descripción:	El padre podrá ver y editar los datos de su hijo	
Actores:	Padre	
Precondiciones:	Ninguna	
Requisitos no funcionales:	Ninguno	
Flujo de eventos:	1. El padre pulsa "Mi hijo" 2. Se cargará un menú [Si pulsa "Ver datos"] 2a. Le aparecerá una pantalla con los datos de su hijo [Si pulsa "Editar datos"] 2b. El padre podrá rellenar los datos que quiera cambiar y pulsa "Editar" [Si los campos introducidos no tienen el formato adecuado] 2b1. Aparecerá un mensaje de error [Si todo es correcto] 2b2. Se modificarán los datos	
Postcondiciones:	Se modificarán los datos del alumno en nuestro sistema	

Tabla 62 - Caso de uso Padre - Editar datos hijo



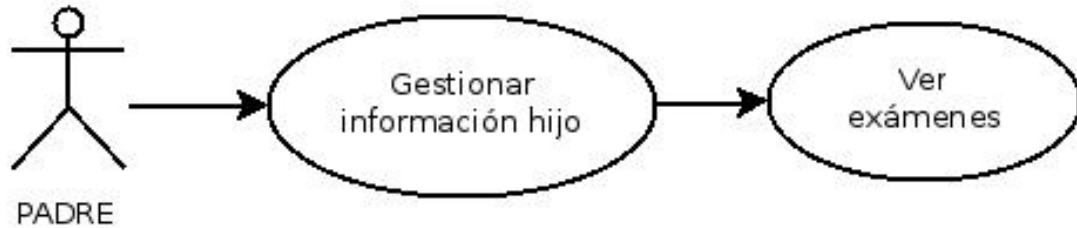
Nombre:	Realizar justificante
Descripción:	El padre podrá realizar justificantes a las faltas de asistencia de su hijo
Actores:	Padre
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Mi hijo" 2. Se cargará un menú [Si pulsa "Ver faltas"] <ol style="list-style-type: none"> 2a. Le parecerá un listado con las faltas de asistencia y si están justificadas 2b. Seleccionamos una falta sin justificar y pulsamos "Realizar justificante" 2c. Rellenamos el justificante y pulsamos "Firmar" <ol style="list-style-type: none"> 2c1. Le aparecerá un mensaje de confirmación y pulsamos "Aceptar" para finalizar la firma
Postcondiciones:	Se justificará la falta en nuestro sistema

Tabla 63 - Caso de uso Padre - Firmar justificación



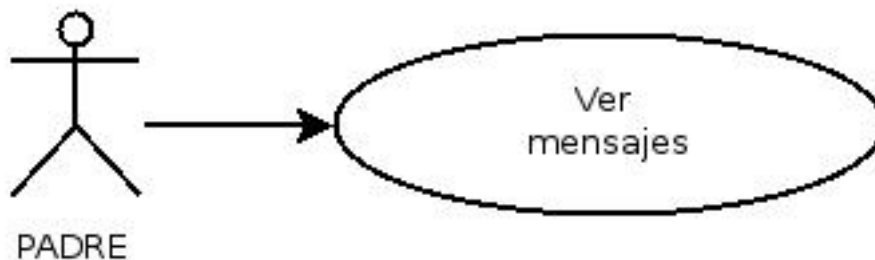
Nombre:	Ver deberes
Descripción:	El padre podrá ver los deberes de su hijo
Actores:	Padre
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Mi hijo" 2. Se cargará un menú [Si pulsa "Ver deberes"] <ol style="list-style-type: none"> 2a. Le parecerá un listado con los deberes pendientes, de que asignatura y que tarea tiene que realizar
Postcondiciones:	Ninguna

Tabla 64 - Caso de uso Padre - Ver deber



Nombre:	Ver exámenes
Descripción:	El padre podrá ver los exámenes de su hijo
Actores:	Padre
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Mi hijo" 2. Se cargará un menú [Si pulsa "Ver exámenes"] <ol style="list-style-type: none"> 2a. Le aparecerá una lista con los exámenes realizados o por realizar con la nota si ya estuvieran corregidos
Postcondiciones:	Ninguna

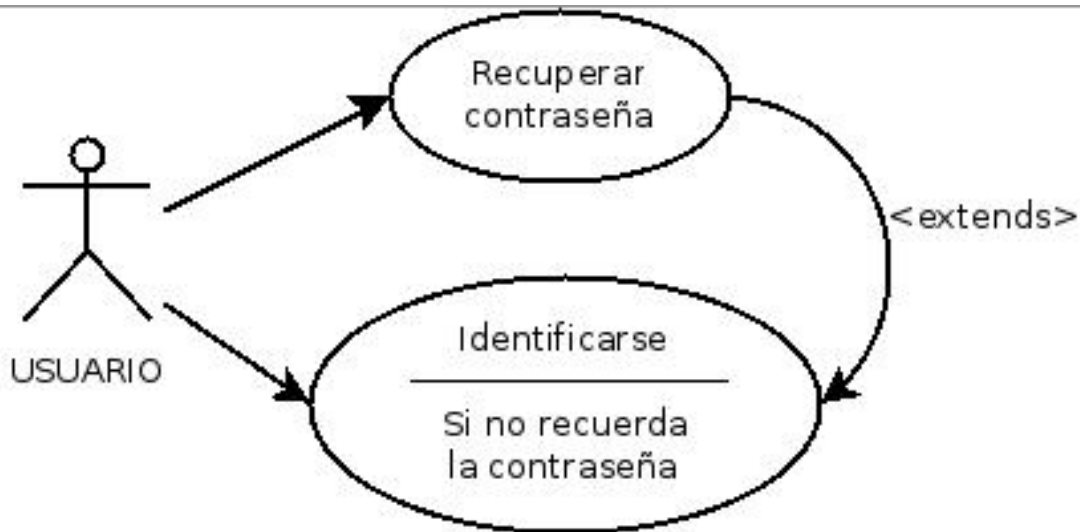
Tabla 65 - Caso de uso Padre - Ver examen



Nombre:	Ver mensajes
Descripción:	El padre podrá tanto ver como enviar mensajes al tutor de su hijo
Actores:	Padre
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El padre pulsa "Ver mensajes" 2. Se cargará una lista con todos los mensajes que tiene [Si selecciona un mensaje y pulsa "Ver mensaje"] <ol style="list-style-type: none"> 2a. Le aparecerá el mensaje [Si selecciona "Mensaje nuevo"] 2b. Rellenará el mensaje y pulsa "Enviar"
Postcondiciones:	Ninguna

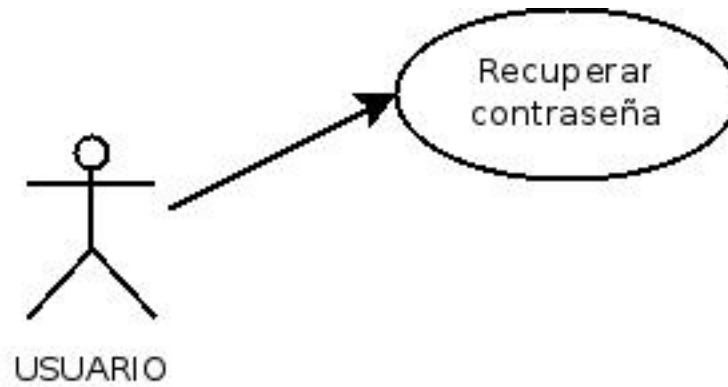
Tabla 66 - Caso de uso Padre - Ver mensajes

12.2.5 Casos de uso del usuario



Nombre:	Identificar usuario
Descripción:	El usuario se identificará para acceder a sus opciones
Actores:	Usuario
Precondiciones:	Los usuarios están registrados automáticamente en el sistema en cuanto se realiza la matriculación del alumno
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. [Si el usuario no recuerda la contraseña y pulsa "He olvidado mi contraseña"] <ol style="list-style-type: none"> 1a. EXTEND Recuperar contraseña (Ver CU "Recupera Contraseña) 2. El usuario introduce el DNI y la contraseña y pulsa "Entrar" 3. Si el DNI y/o la contraseña no son correctos, se muestra un mensaje de error y vuelve a la pantalla anterior
Postcondiciones:	Se identificará el usuario si introduce los datos correctamente, si no le aparecerá un mensaje de error

Tabla 67 - Caso de uso Usuario - Identificarse



Nombre:	Recuperar contraseña
Descripción:	Se enviará una contraseña al email del usuario, con DNI asociado, generada de manera aleatoria.
Actores:	Usuario
Precondiciones:	Ninguna
Requisitos no funcionales:	Ninguno
Flujo de eventos:	<ol style="list-style-type: none"> 1. El usuario hace click en "He olvidado mi contraseña" 2. El usuario introduce su DNI 3. Aparecerá un aviso de confirmación
Postcondiciones:	El usuario recibirá en su correo un email con una nueva contraseña

Tabla 68 - Caso de uso Usuario - Recuperar contraseña

12.3 Modelo de Dominio

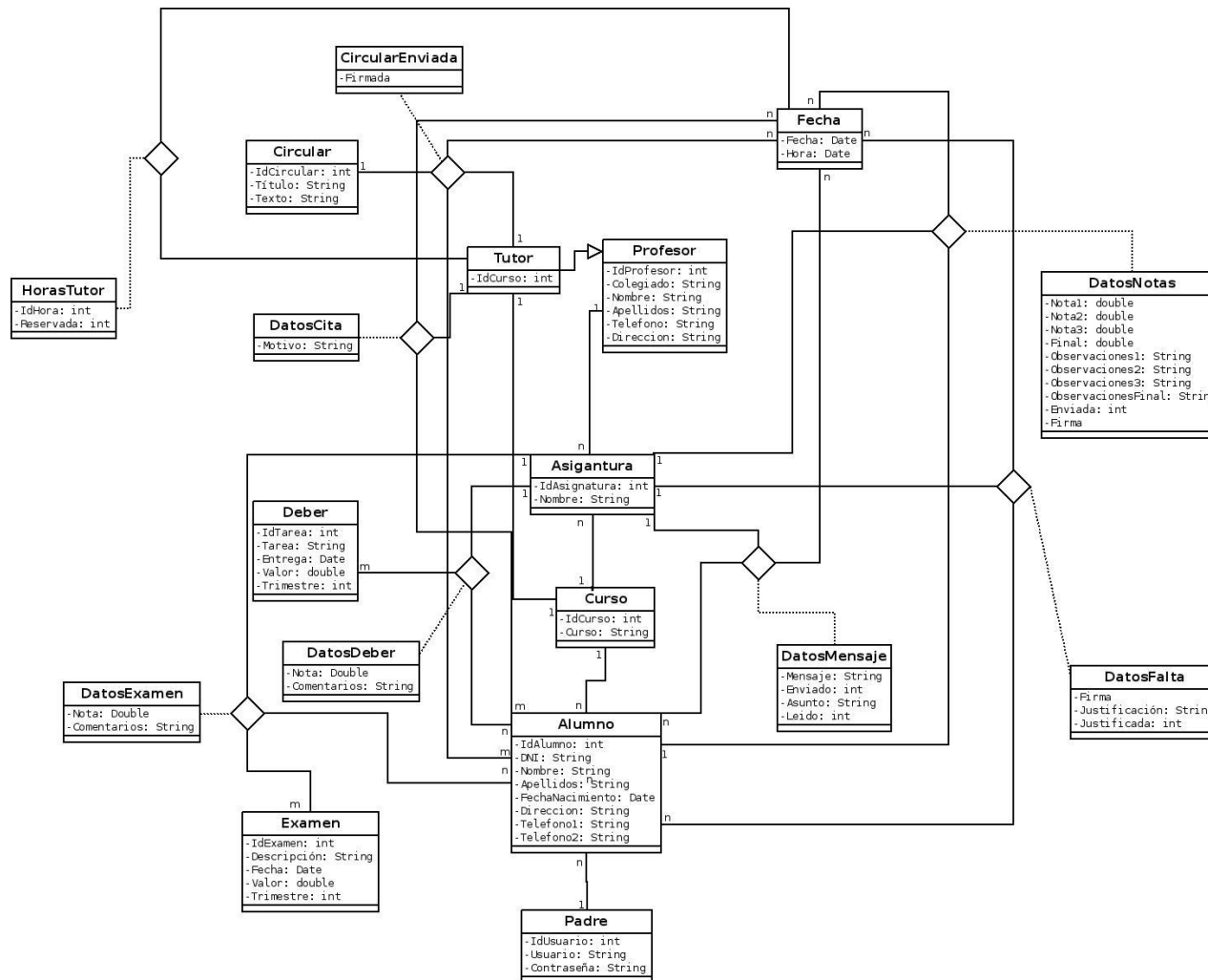


Ilustración 13 - Modelo de Dominio

12.5 Diagrama Entidad Relación Base de Datos

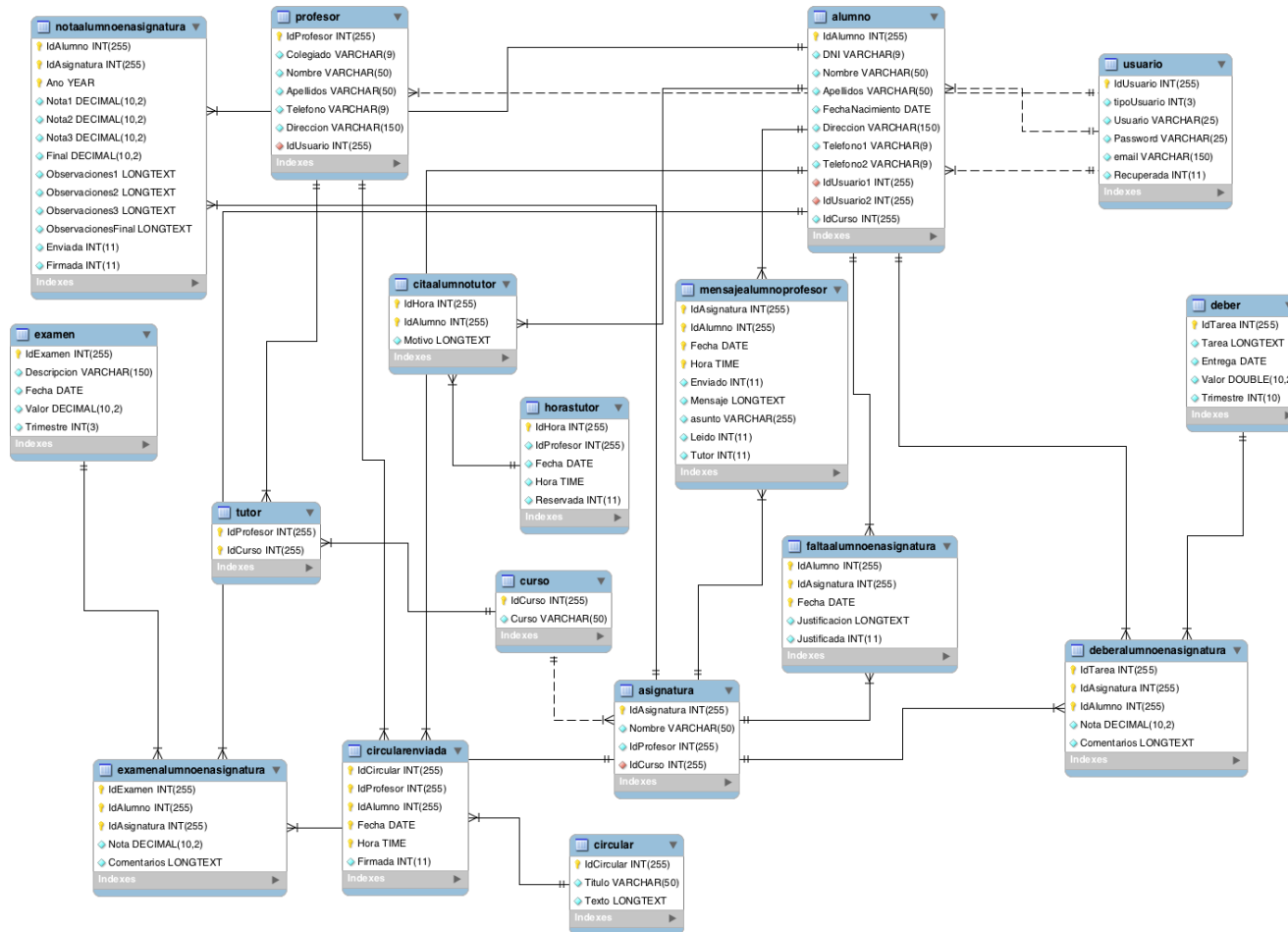
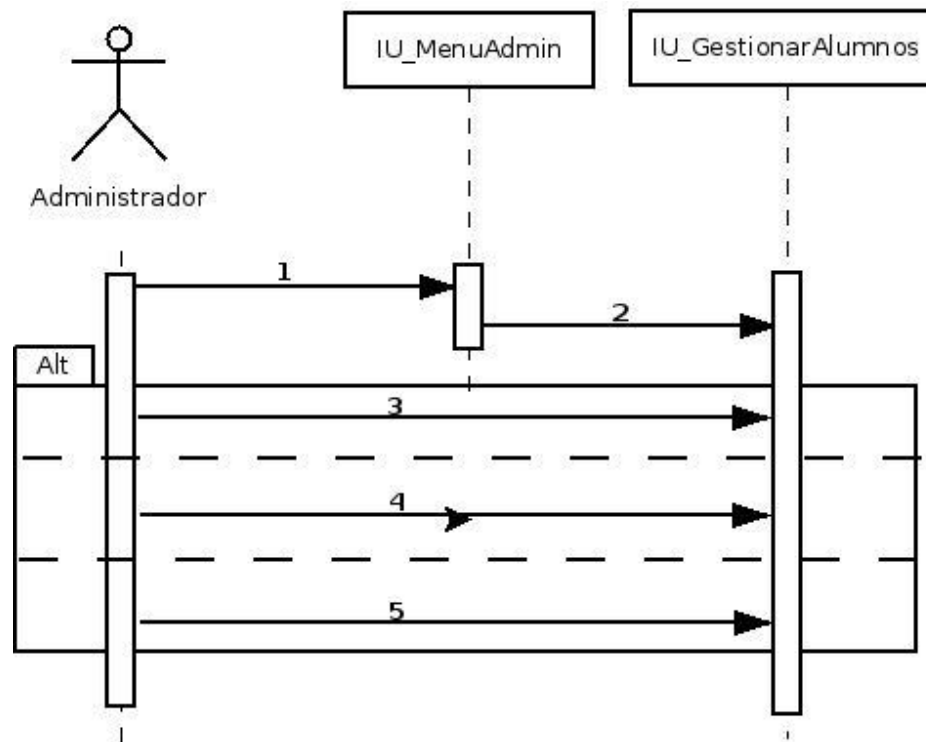


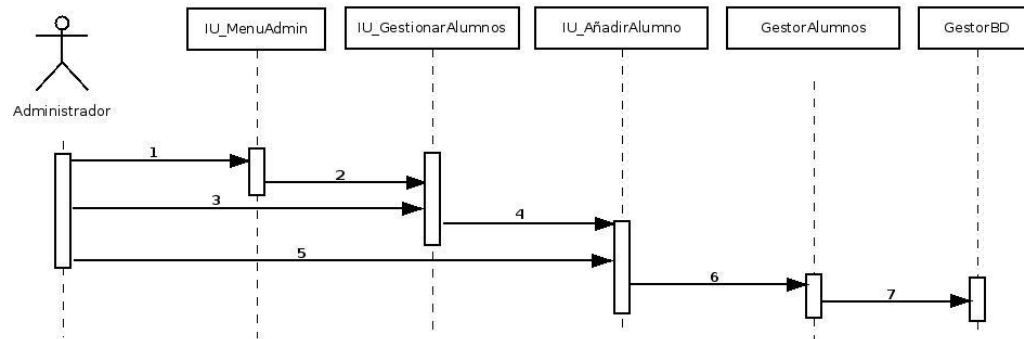
Ilustración 15 - Diagrama Entidad-Relación

12.5.1 Diagramas de secuencia del Administrador



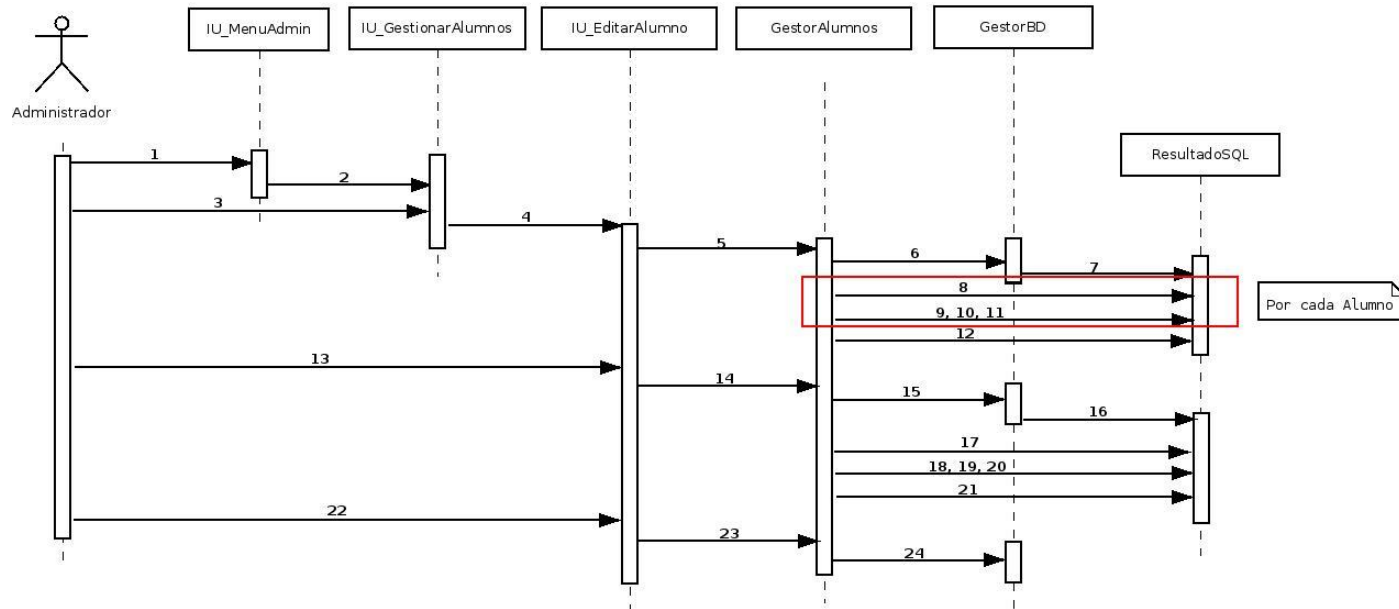
1. El administrador pulsa el botón "Gestionar alumnos"
2. new()
3. Si el administrador pulsa el botón "Añadir alumno" (SUBCASO AñadirAlumno)
4. Si el administrador pulsa el botón "Editar alumno" (SUBCASO EditarAlumno)
5. Si el administrador pulsa el botón "Eliminar alumno" (SUBCASO EliminarAlumno)

Ilustración 16 Diagrama Gestionar Alumno



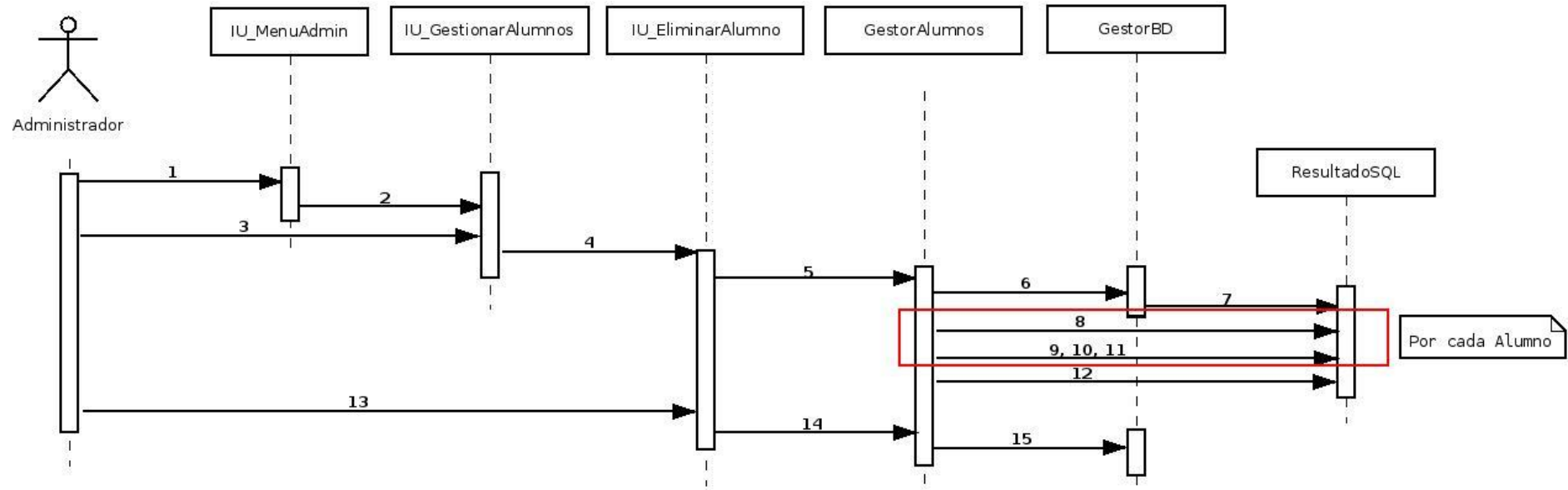
1. El administrador pulsa el botón "Gestionar alumnos"
2. new()
3. El administrador pulsa el botón "Añadir alumno"
4. new()
5. El usuario rellena los datos y pulsa "Añadir"
6. añadirAlumno(DNI, Nombre, Apellidos, FechaNacimiento, Dirección, Teléfono, Usuario, Curso)
7. execSQL("INSERT INTO Alumno (DNI, Nombre, Apellidos, FechaNacimiento, Dirección, Teléfono, idUsuario, IdCurso) VALUES (%DNI%, %Nombre%, %Apellidos%, %FechaNacimiento%, %Dirección%, %Teléfono%, %Usuario%, %Curso%)")

Ilustración 17 - Diagrama Añadir alumno



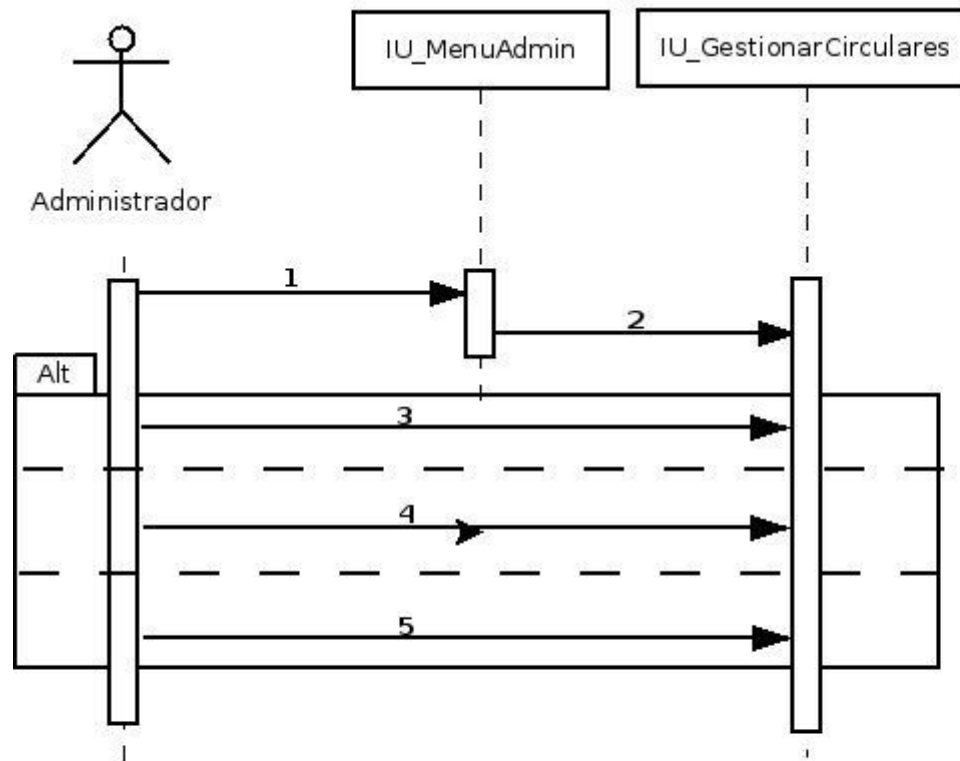
1. El administrador pulsa el botón "Gestionar alumnos"
2. new()
3. El administrador pulsa el botón "Editar alumno"
4. new()
5. cargarAlumnos() : ListaAlumnos
6. execSQL("SELECT * FROM Alumno")
7. new()
8. next()
9. getInteger("IdAlumno") : Integer
10. getString("Apellidos") : String
11. getString("Nombre") : String
12. close()
13. El administrador selecciona un alumno
14. mostrarDatosAlumno("IdAlumnoSeleccionado")
15. execSQL("SELECT * FROM Alumno WHERE IdAlumno=%IdAlumnoSeleccionado%")
16. new()
17. next()
18. getString("Dirección") : String
19. getString("Teléfono") : String
20. getInteger("IdCurso") : Integer
21. close()
22. El administrador modifica los datos y pulsa "Editar"
23. modificarAlumno(IdAlumno, Dirección, Teléfono, IdCurso)
24. execSQL("UPDATE Alumno SET Dirección=%Dirección%, Teléfono=%Teléfono%, IdCurso=%IdCurso% WHERE IdAlumno=%IdAlumnoS eleccionado%")

Ilustración 18 - Diagrama Editar Alumno



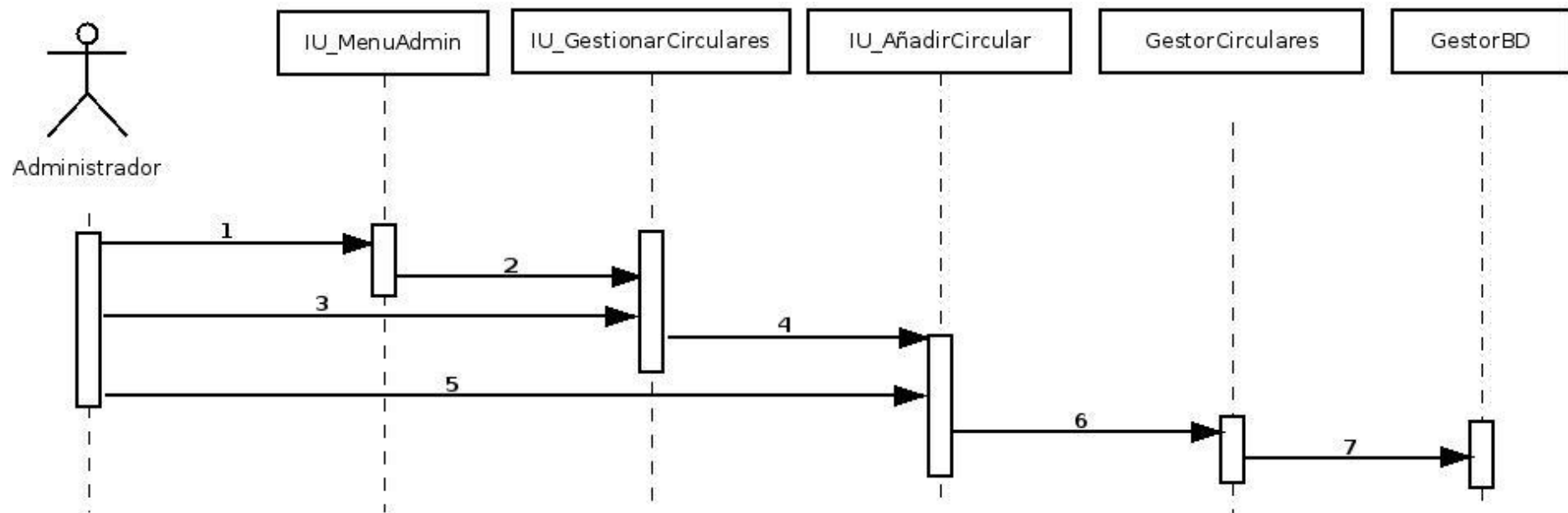
1. El administrador pulsa el botón "Gestionar alumnos"
2. new()
3. El administrador pulsa el botón "Eliminar alumno"
4. new()
5. cargarAlumnos() : ListaAlumnos
6. execSQL("SELECT * FROM Alumno")
7. new()
8. next()
9. getInteger("IdAlumno") : Integer
10. getString("Apellidos") : String
11. getString("Nombre") : String
12. close()
13. El administrador selecciona un alumno
14. eliminarAlumno("IdAlumnoSeleccionado")
15. execSQL("DELETE FROM Alumno WHERE IdAlumno=%IdAlumnoSeleccionado")

Ilustración 19 - Diagrama Eliminar Alumno



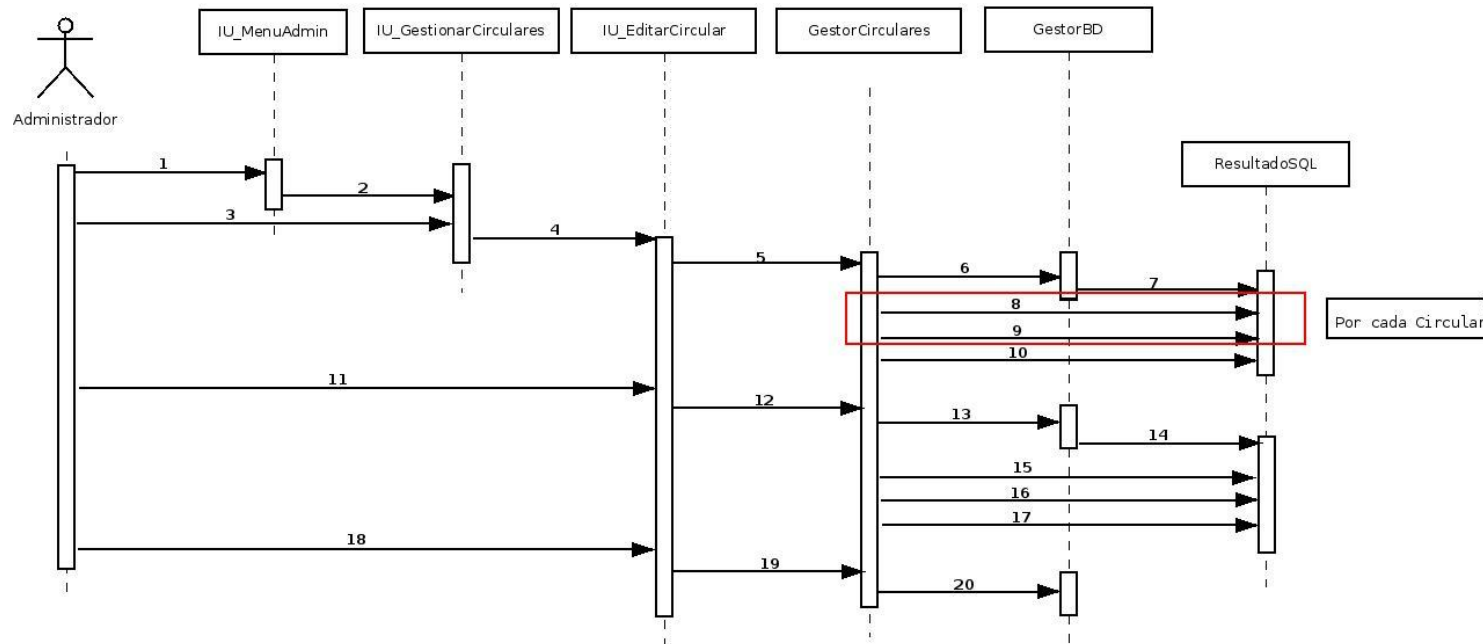
1. El administrador pulsa el botón "Gestionar circulares"
2. new()
3. Si el administrador pulsa el botón "Añadir circular" (SUBCASO AñadirCircular)
4. Si el administrador pulsa el botón "Editar circular" (SUBCASO EditarCircular)
5. Si el administrador pulsa el botón "Eliminar circular" (SUBCASO EliminarCircular)

Ilustración 20 - Diagrama Gestionar Circulares



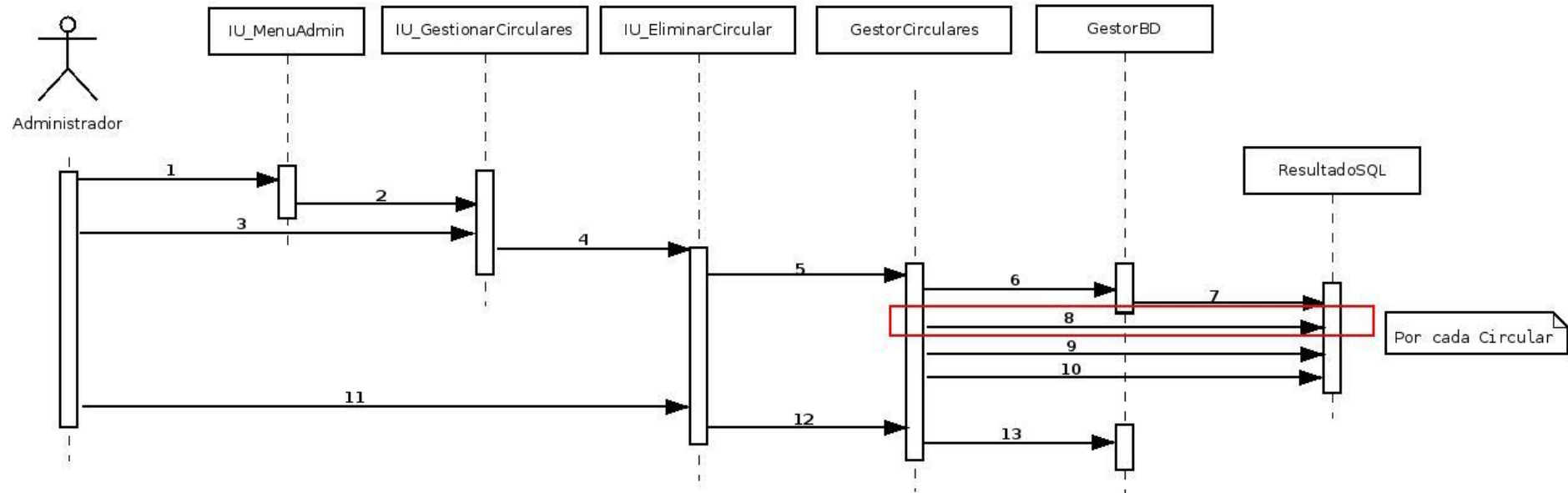
1. El administrador pulsa el botón "Gestionar circulares"
2. new()
3. El administrador pulsa el botón "Añadir circular"
4. new()
5. El usuario rellena los datos y pulsa "Añadir"
6. añadirCircular(Texto)
7. execSQL("INSERT INTO Circular (Texto) VALUES (%Text o%)")

Ilustración 21 - Diagrama Añadir Circular



1. El administrador pulsa el botón "Gestionar circulares"
2. new()
3. El administrador pulsa el botón "Editar circular"
4. new()
5. cargarCirculares()
6. execSQL("SELECT * FROM Circular")
7. new()
8. next()
9. getString("IdCircular") : String
10. close()
11. El administrador selecciona una circular
12. mostrarCircular("IdCircularSeleccionada")
13. execSQL("SELECT * FROM Alumno WHERE IdCircular=%IdCircularSeleccionada%")
14. new()
15. next()
16. getString("Texto") : String
17. close()
18. El administrador modifica los datos y pulsa "Editar"
19. modificarCircular(IdCircular, Texto)
20. execSQL("UPDATE Circular SET Texto=%Texto% WHERE idCircular=%IdCircularSeleccionado%")

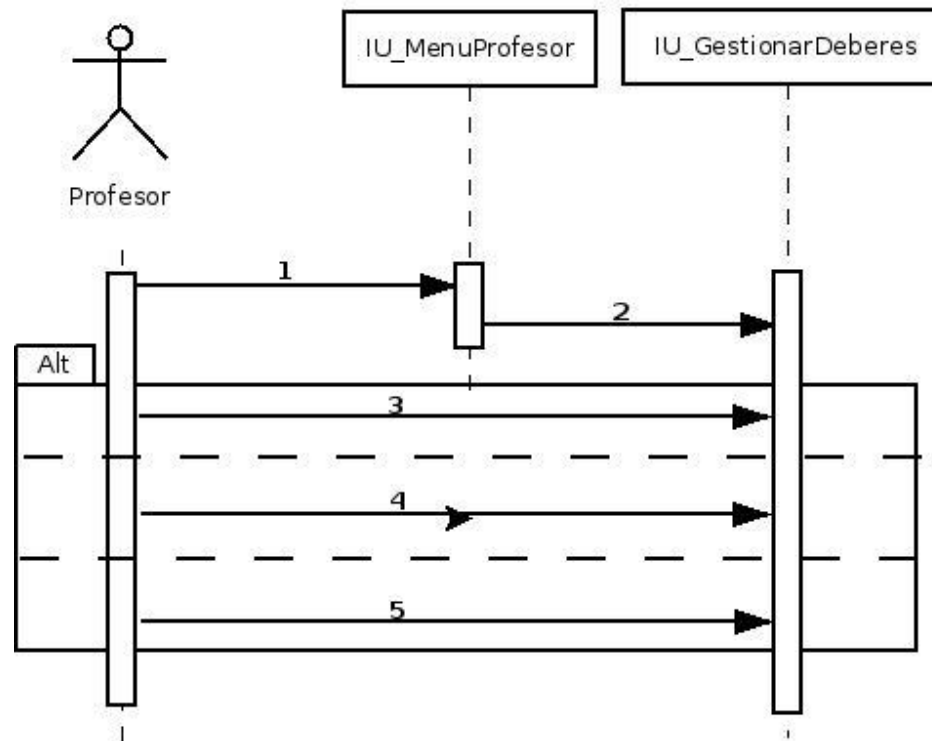
Ilustración 22 - Diagrama Editar Circular



1. El administrador pulsa el botón "Gestionar circulares"
2. new()
3. El administrador pulsa el botón "Eliminar circular"
4. new()
5. cargarCircular()
6. execSQL("SELECT * FROM Circular")
7. new()
8. next()
9. getInteger("IdCircular") : Integer
10. close()
11. El administrador selecciona una circular
12. eliminarCircular("IdCircularSeleccionada")
13. execSQL("DELETE FROM Circular WHERE IdCircular=%IdCircularSeleccionada%")

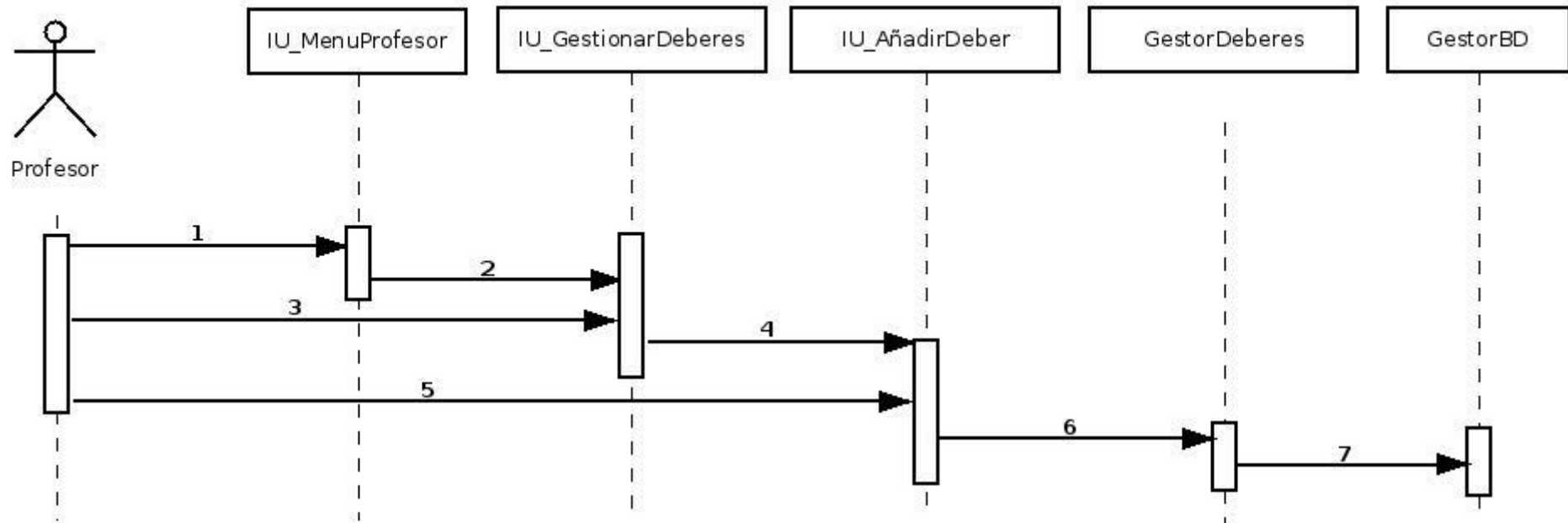
Ilustración 23 - Diagrama Eliminar Circular

12.5.2 Diagramas de secuencia de Profesor



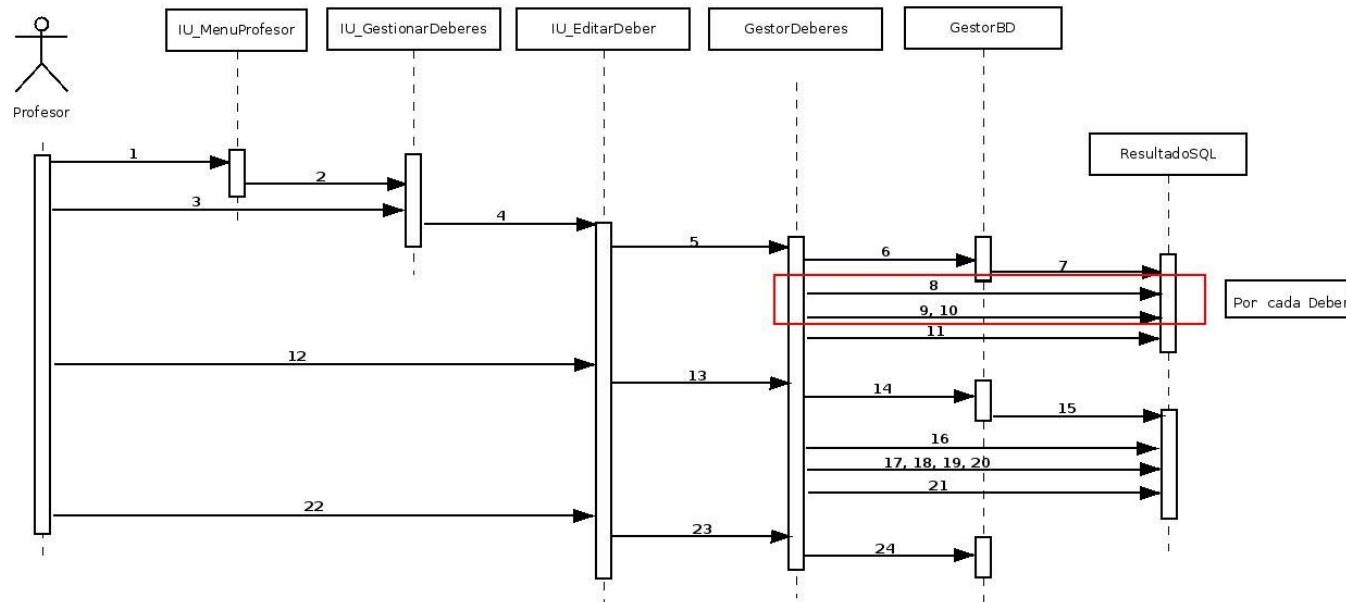
1. El profesor pulsa el botón "Gestionar deberes"
2. new()
3. Si el profesor pulsa el botón "Añadir deber" (SUBCASO AñadirDeber)
4. Si el profesor pulsa el botón "Editar deber" (SUBCASO EditarDeber)
5. Si el profesor pulsa el botón "Eliminar deber" (SUBCASO EliminarDeber)

Ilustración 24 - Diagrama Gestionar Deberes



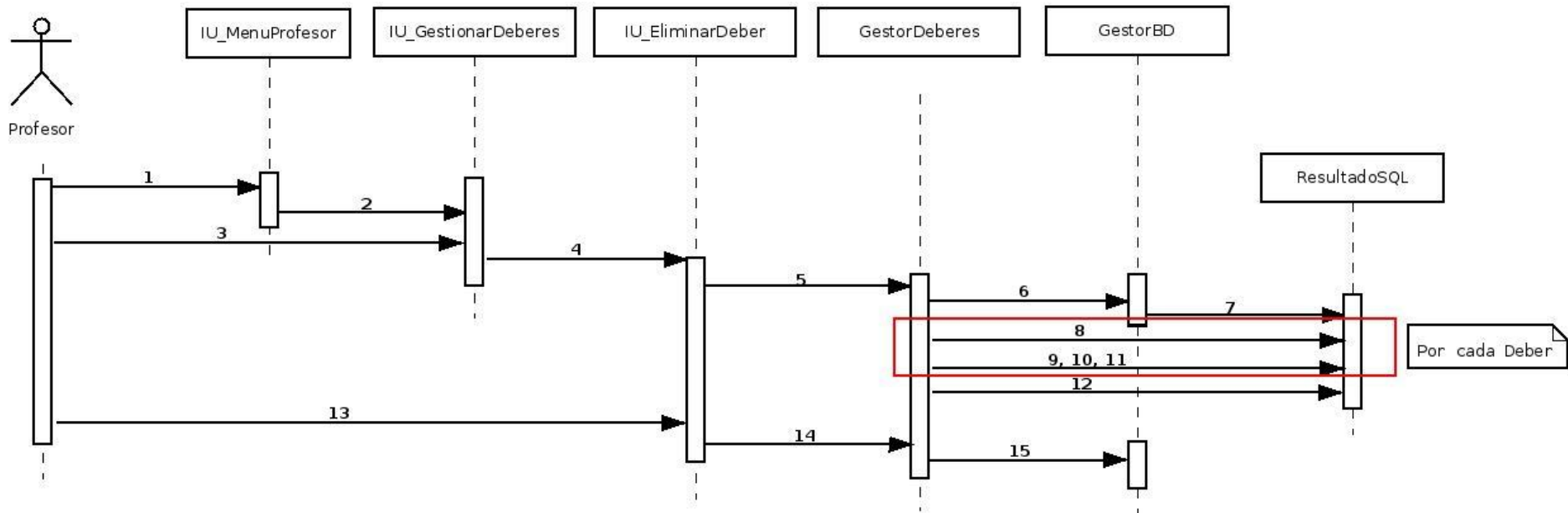
1. El profesor pulsa el botón "Gestionar deberes"
2. new()
3. El profesor pulsa el botón "Añadir deber"
4. new()
5. El usuario rellena los datos y pulsa "Añadir"
6. añadirDeber(Tarea, Entrega, Valor, Trimestre)
7. execSQL("INSERT INTO Deber (Tarea, Entrega, Valor, Trimestre) VALUES (%Tarea%, %Entrega%, %Valor%, %Trimestre%)")

Ilustración 25 - Diagrama Añadir Deber



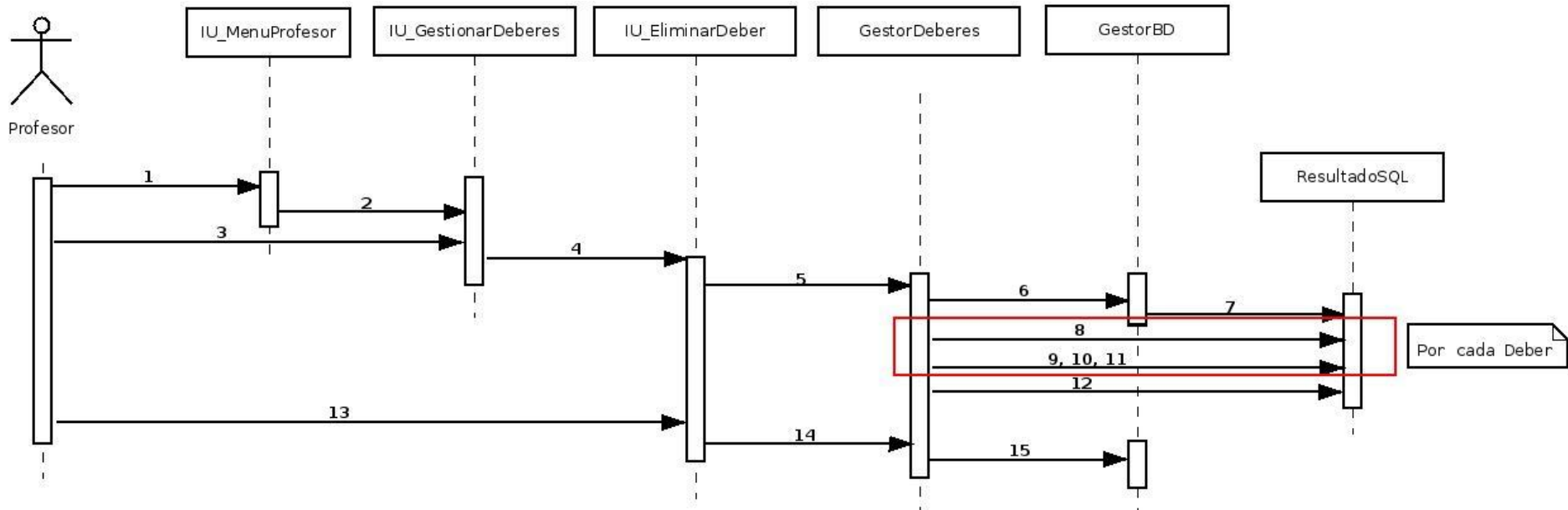
1. El profesor pulsa el botón "Gestionar deberes"
2. new()
3. El profesor pulsa el botón "Editar deber"
4. new()
5. cargarDeberes() : ListaDeberes
6. execSQL("SELECT * FROM Deber")
7. new()
8. next()
9. getInteger("IdTarea") : Integer
10. getString("Tarea") : String
11. close()
12. El profesor selecciona un deber
13. mostrarDatosDeber("IdTareaSeleccionada")
14. execSQL("SELECT * FROM Deber WHERE IdTarea=%IdTareaSeleccionada%")
15. new()
16. next()
17. getString("Tarea") : String
18. getDate("Entrega") : Date
19. getDouble("Valor") : Double
20. getInteger("Trimestre") : Integer
21. close()
22. El profesor modifica los datos y pulsa "Editar"
23. modificarDeber(IdTareaSeleccionada, Tarea, Entrega, Valor, Trimestre)
24. execSQL("UPDATE Deber SET Tarea=%Tarea%, Entrega=%Entrega%, Valor=%Valor%, Trimestre=%Trimestre% WHERE idTarea=%IdTareaSeleccionada%")

Ilustración 26 - Diagrama Editar Deber



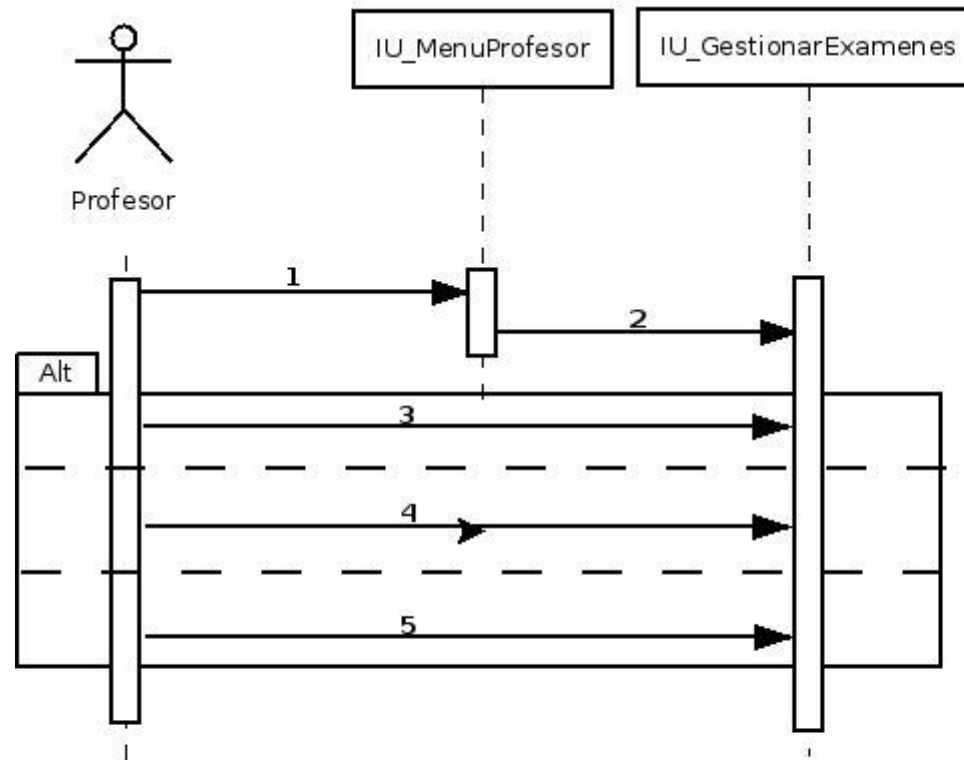
1. El profesor pulsa el botón "Gestionar deberes"
2. new()
3. El profesor pulsa el botón "Eliminar deber"
4. new()
5. cargarDeberes() : ListaDeberes
6. execSQL("SELECT * FROM Deber")
7. new()
8. next()
9. getInteger("IdTarea") : Integer
10. getString("Tarea") : String
11. getDate("Entrega") : Date
12. close()
13. El profesor selecciona un deber
14. eliminarDeber("IdTareaSeleccionada")
15. execSQL("DELETE FROM Deber WHERE IdTarea=%IdTareaSeleccionada%")

Ilustración 27 - Diagrama Eliminar Deber



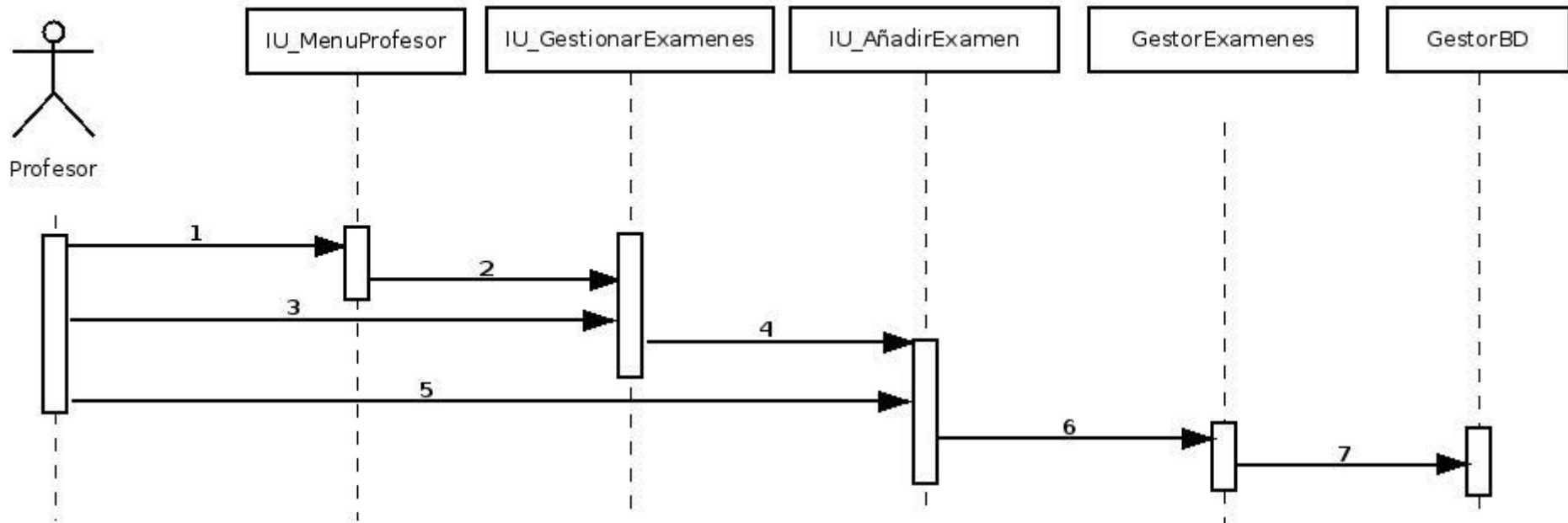
1. El profesor pulsa el botón "Gestionar deberes"
2. `new()`
3. El profesor pulsa el botón "Eliminar deber"
4. `new()`
5. `cargarDeberes() : ListaDeberes`
6. `execSQL("SELECT * FROM Deber")`
7. `new()`
8. `next()`
9. `getInteger("IdTarea") : Integer`
10. `getString("Tarea") : String`
11. `getDate("Entrega") : Date`
12. `close()`
13. El profesor selecciona un deber
14. `eliminarDeber("IdTareaSeleccionada")`
15. `execSQL("DELETE FROM Deber WHERE IdTarea=%dIdTareaSeleccionada%")`

Ilustración 28 - Diagrama Eliminar Deber



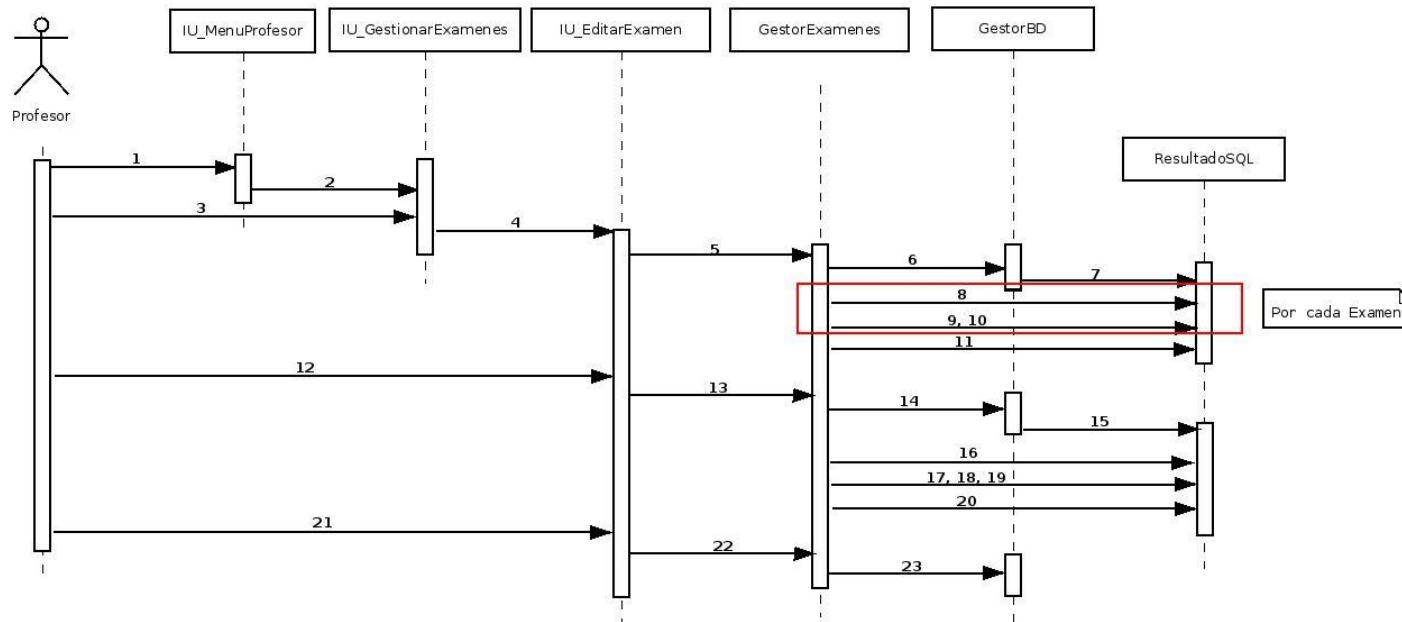
1. El profesor pulsa el botón "Gestionar exámenes"
2. new()
3. Si el profesor pulsa el botón "Añadir examen" (SUBCASO AñadirExamen)
4. Si el profesor pulsa el botón "Editar examen" (SUBCASO EditarExamen)
5. Si el profesor pulsa el botón "Eliminar examen" (SUBCASO EliminarExamen)

Ilustración 29 - Diagrama Gestionar Exámenes



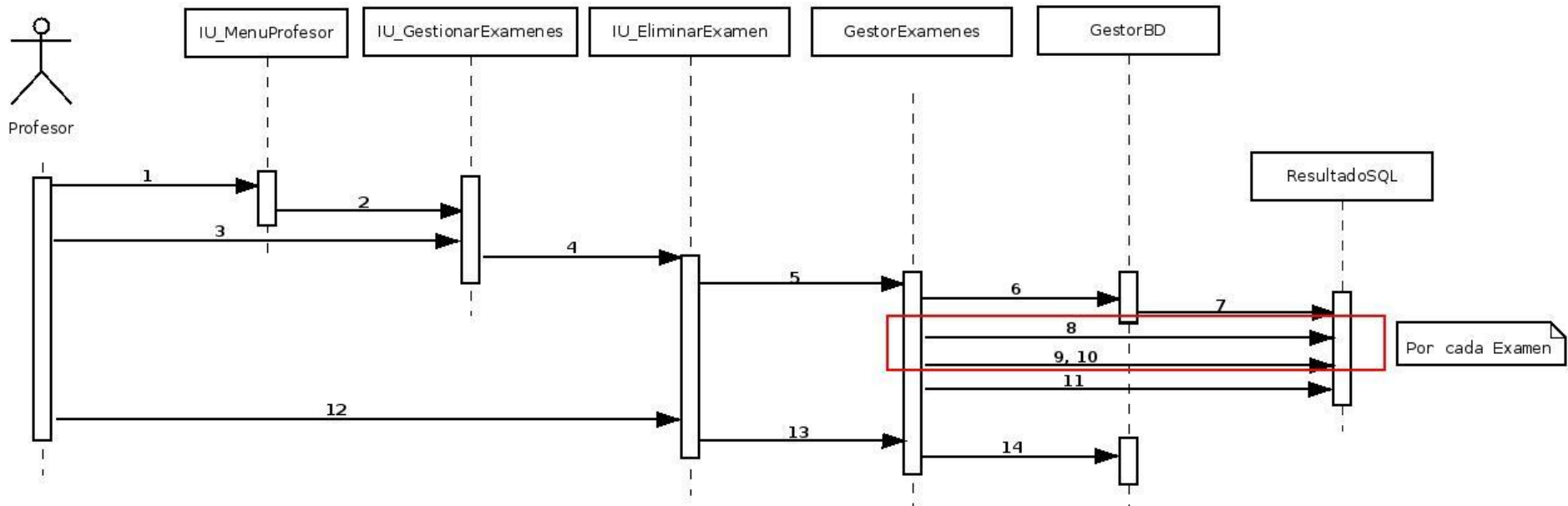
1. El profesor pulsa el botón "Gestionar exámenes"
2. new()
3. El profesor pulsa el botón "Añadir examen"
4. new()
5. El usuario rellena los datos y pulsa "Añadir"
6. añadirExamen(Nombre, Valor, Trimestre)
7. execSQL("INSERT INTO Examen (Nombre, Valor, Trimestre) VALUES (%Nombre%, %Valor%, %Trimestre%)")

Ilustración 30 - Diagrama Añadir Examen



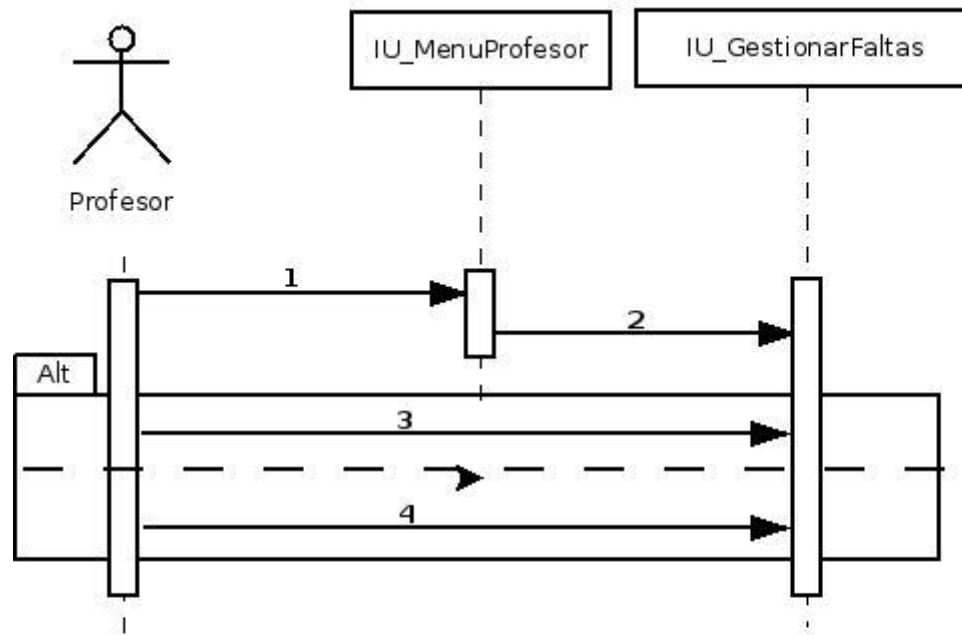
1. El profesor pulsa el botón "Gestionar exámenes"
2. new()
3. El profesor pulsa el botón "Editar examen"
4. new()
5. cargarExámenes(): ListaExámenes
6. execSQL("SELECT * FROM Examen")
7. new()
8. next()
9. getInteger("IdExamen"): Integer
10. getString("Nombre"): String
11. close()
12. El profesor selecciona un examen
13. mostrarDatosExamen("IdExamenSeleccionado")
14. execSQL("SELECT * FROM Examen WHERE IdExamen=%IdExamenSeleccionado%")
15. new()
16. next()
17. getString("Nombre"): String
18. getDouble("Valor"): Double
19. getInteger("Trimestre"): Integer
20. close()
21. El profesor modifica los datos y pulsa "Editar"
22. modificarExamen(IdExamenSeleccionado, Nombre, Valor, Trimestre)
23. execSQL("UPDATE Examen SET Nombre=%Nombre%, Valor=%Valor%, Trimestre=%Trimestre% WHERE idExamen=%IdExamenSeleccionado%")

Ilustración 31 - Diagrama Editar Examen



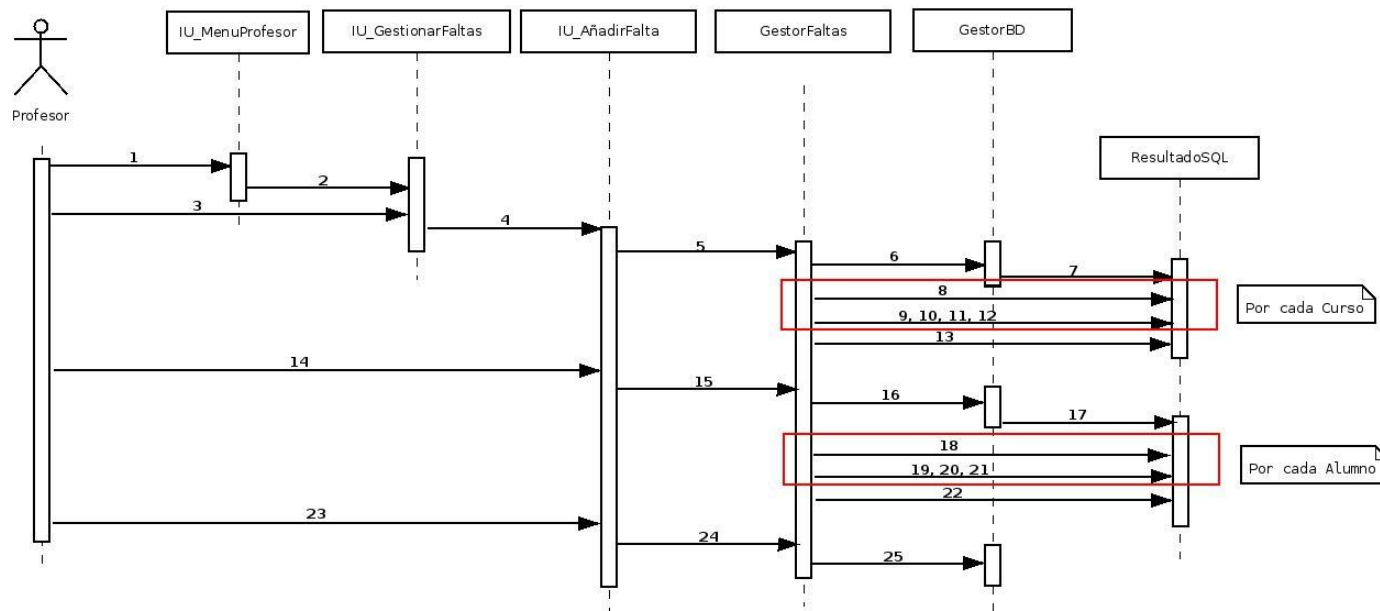
1. El profesor pulsa el botón "Gestionar exámenes"
2. `new()`
3. El profesor pulsa el botón "Eliminar examen"
4. `new()`
5. `cargarExámenes() : ListaExámenes`
6. `execSQL("SELECT * FROM Examen")`
7. `new()`
8. `next()`
9. `getInteger("IdExamen") : Integer`
10. `getString("Nombre") : String`
11. `close()`
12. El profesor selecciona un deber
13. `eliminarDeber("IdExamenSeleccionado")`
14. `execSQL("DELETE FROM Examen WHERE IdExamen=%IdExamenSeleccionado%")`

Ilustración 32 - Diagrama Eliminar Examen



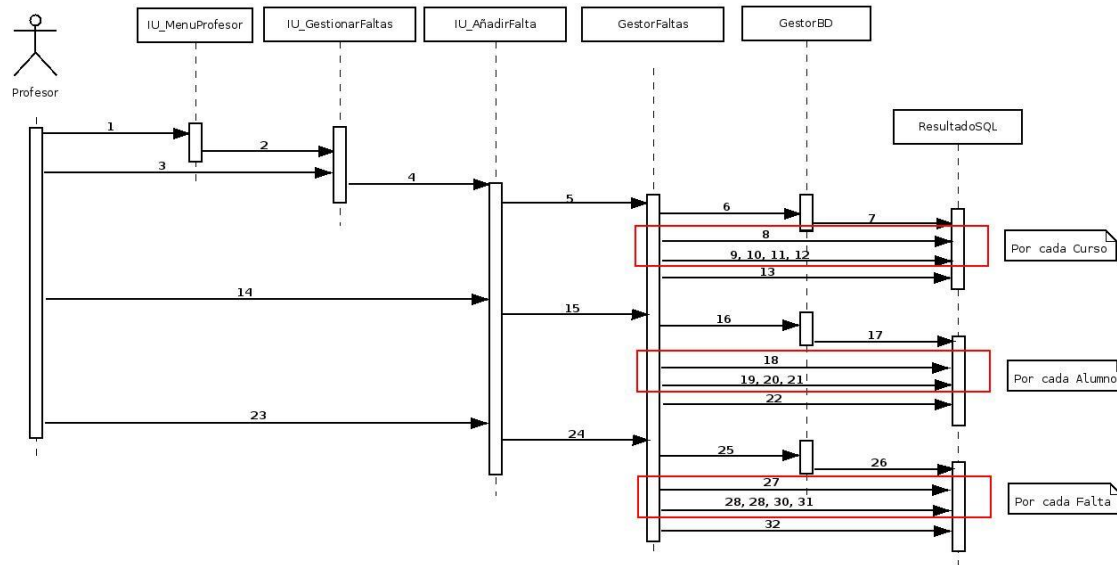
1. El profesor pulsa el botón "Gestionar faltas"
2. `new()`
3. Si el profesor pulsa el botón "Añadir falta" (SUBCASO AñadirFalta)
4. Si el profesor pulsa el botón "Ver falta" (SUBCASO VerFaltas)

Ilustración 33 - Diagrama Gestionar Faltas



1. El profesor pulsa el botón "Ver info alumnos"
2. new()
3. El profesor pulsa el botón "Añadir falta"
4. new()
5. cargarCursos(UsuarioRegistrado) : ListaCursos()
6. `execSQL("SELECT * FROM AsignaturaImpartida, Asignatura, Curso WHERE Colegiado=%UsuarioRegistrado% AND Curso.IdCurso=AsignaturaImpartida.IdCurso AND AsignaturaImpartida.IdAsignatura=Asignatura.IdCurso")`
7. new()
8. next()
9. `getInteger("IdCurso") : Integer`
10. `getInteger("IdAsignatura") : Integer`
11. `getString("Curso") : String`
12. `getString("Nombre") : String`
13. close()
14. El profesor selecciona un curso
15. cargarAlumnos(IdCursoSeleccionado) : ListaAlumnos
16. `execSQL("SELECT * FROM Alumno WHERE IdCurso=%IdCursoSeleccionado%")`
17. new()
18. next()
19. `getInteger("IdAlumno") : Integer`
20. `getString("Nombre") : String`
21. `getString("Apellidos") : String`
22. close()
23. El profesor selecciona un alumno y selecciona "Añadir Falta"
24. `añadirFalta(IdAlumnoSeleccionado, IdAsignaturaSeleccionada, Fecha, Hora)`
25. `execSQL("INSERT INTO FaltaAlumnoEnAsignatura (IdAlumno, IdAsignatura, Fecha, Hora) VALUES (%IdAlumno%, %IdAsignatura%, %Fecha%, %Hora%)")`

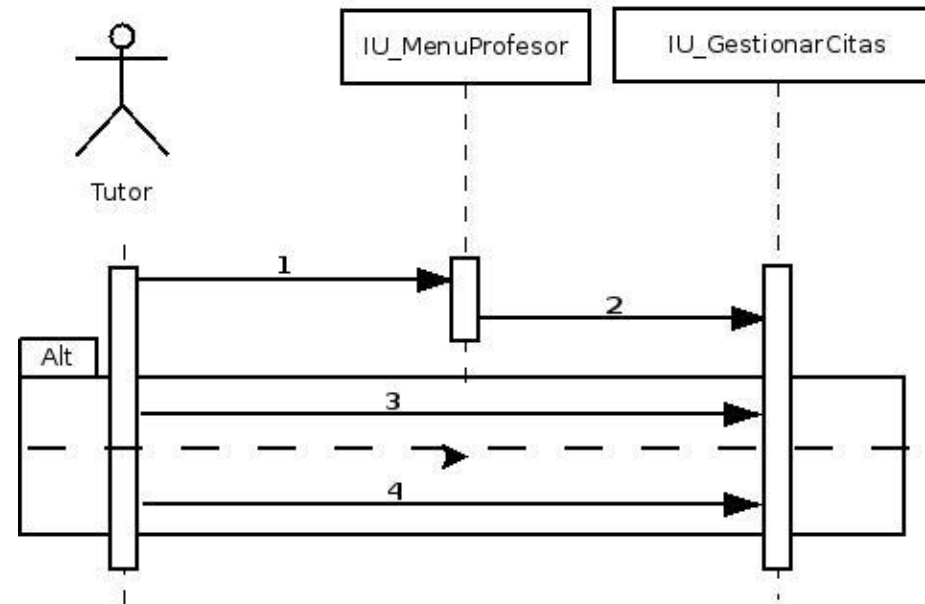
Ilustración 34 - Diagrama Añadir Falta



1. El profesor pulsa el botón "Ver info alumnos"
2. new()
3. El profesor pulsa el botón "Añadir falta"
4. new()
5. cargarCursos(UsuarioRegistrado) : ListaCursos()
6. execSQL("SELECT * FROM AsignaturaImpartida, Asignatura, Curso WHERE Colegiado=%UsuarioRegistrado% AND Curso.IdCurso=AsignaturaImpartida.IdCurso AND AsignaturaImpartida.IdAsignatura=Asignatura.IdCurso")
7. new()
8. next()
9. getInteger("IdCurso") : Integer
10. getInteger("IdAsignatura") : Integer
11. getString("Curso") : String
12. getString("Nombre") : String
13. close()
14. El profesor selecciona un curso
15. cargarAlumnos(IdCursoSeleccionado) : ListaAlumnos
16. execSQL("SELECT * FROM Alumno WHERE IdCurso=%IdCursoSeleccionado%")
17. new()
18. next()
19. getInteger("IdAlumno") : Integer
20. getString("Nombre") : String
21. getString("Apellidos") : String
22. close()
23. El profesor selecciona un alumno y selecciona "Ver Faltas"
24. CargarFaltas(IdAlumnoSeleccionado, IdAsignaturaSeleccionada)
25. execSQL("SELECT * FROM FaltaAlumnoEnAsignatura WHERE IdAlumno=%IdAlumnoSeleccionado% AND IdAsignatura=%IdAsignaturaSeleccionada%")
26. new()
27. next()
28. getDate("Fecha") : Date
29. getDate("Hora") : Date
30. getString("Justificación") : String
31. getString("Firma") : String
32. close()

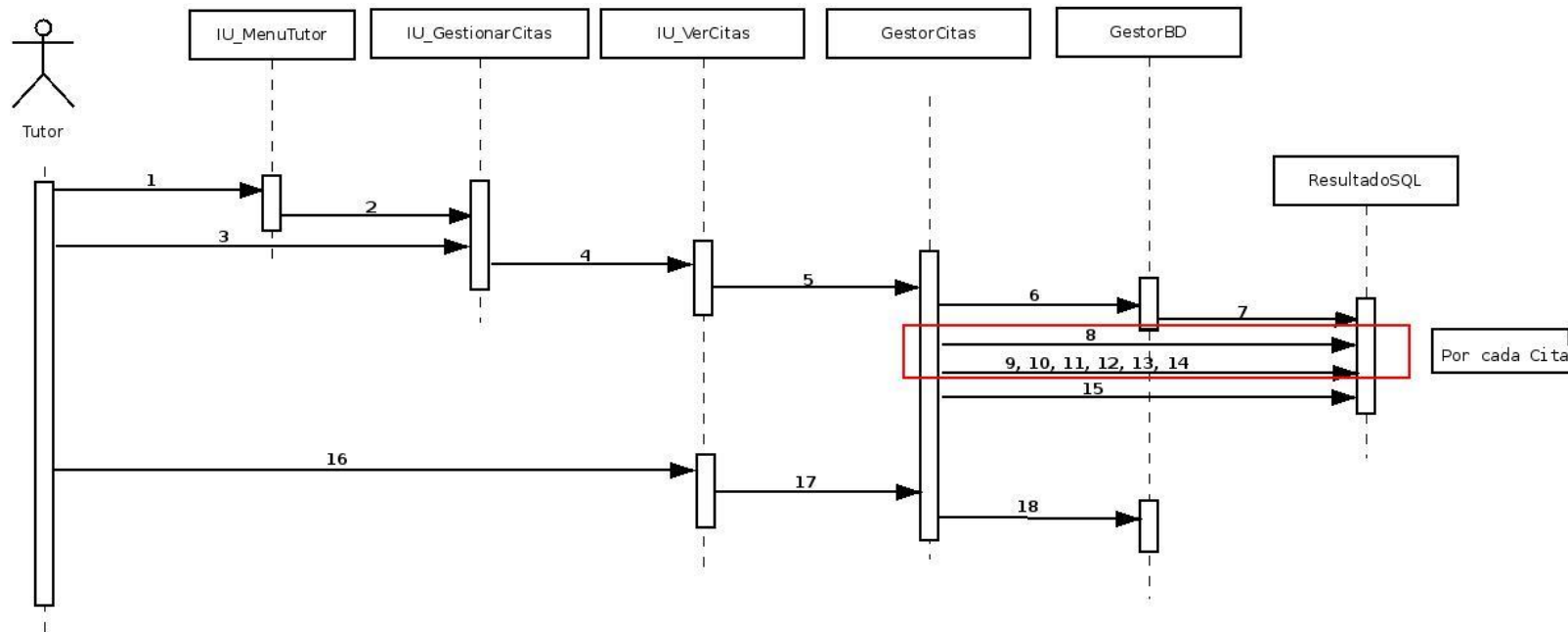
Ilustración 35 - Diagrama Ver Faltas

12.5.3 Diagramas de secuencia de Tutor



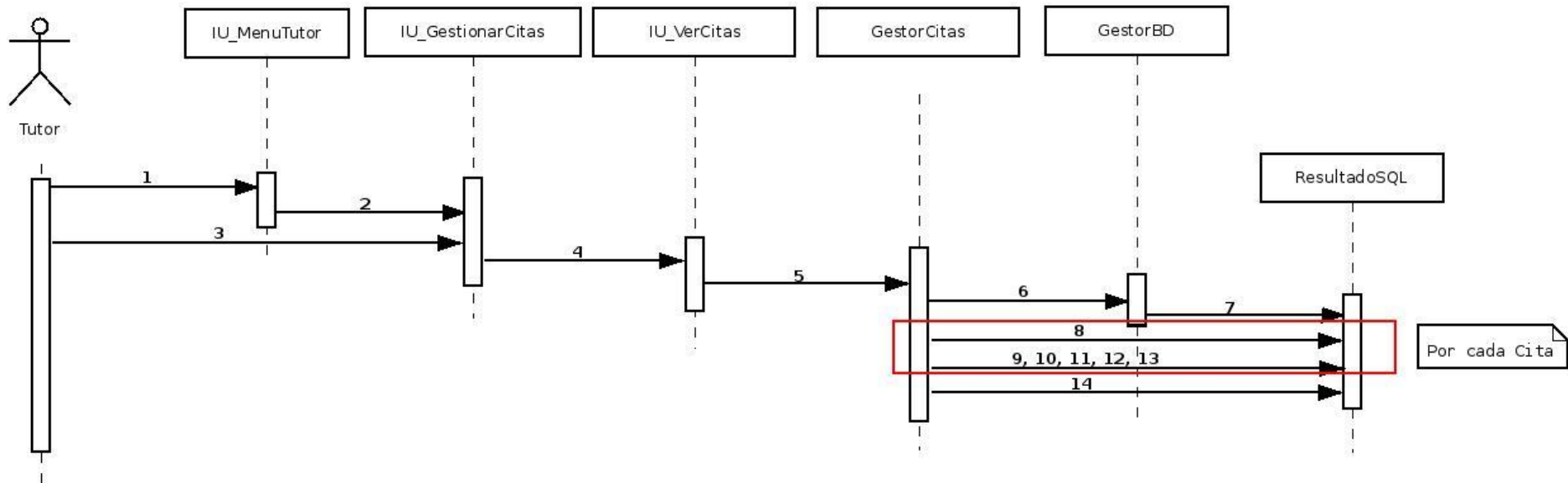
1. El profesor pulsa el botón "Gestionar Citas"
2. new()
3. Si el tutor pulsa el botón "Eliminar cita" (SUBCASO EliminarCitas)
4. Si el tutor pulsa el botón "Ver citas" (SUBCASO VerCitas)

Ilustración 36 - Diagrama Gestionar Citas



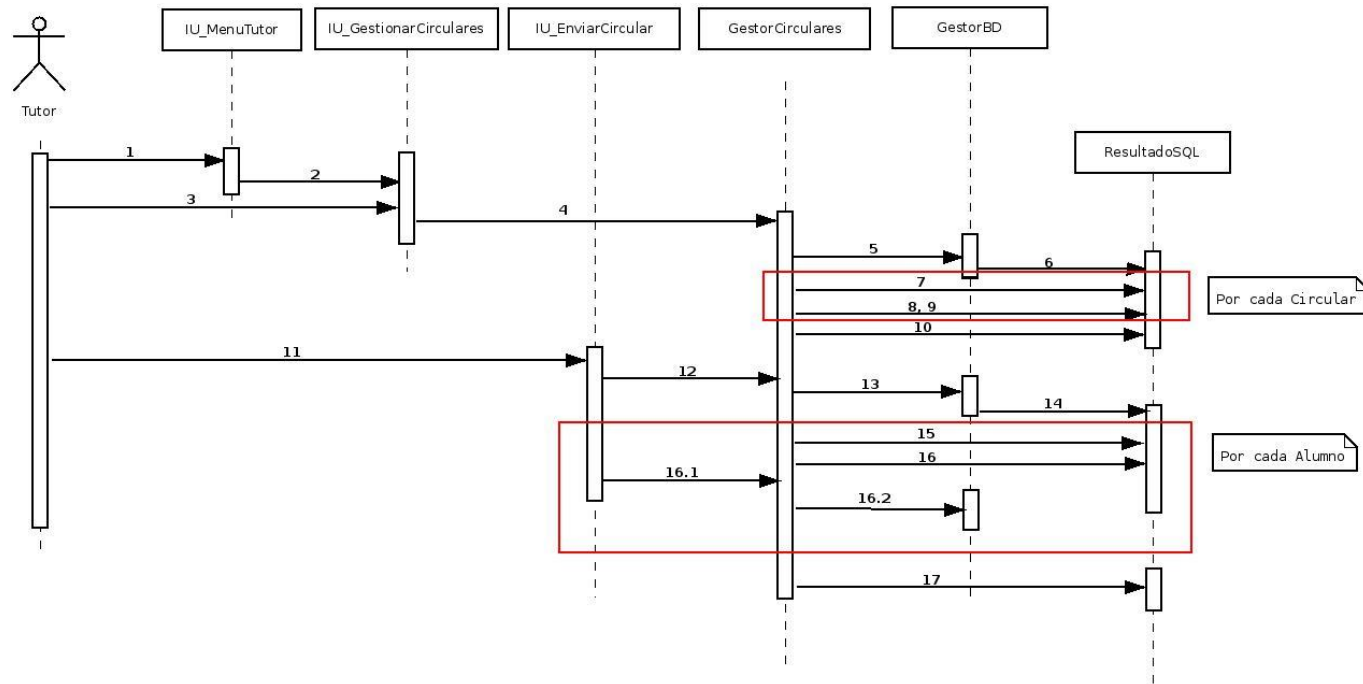
1. El tutor pulsa el botón "Gestionar citas"
2. new()
3. El tutor pulsa el botón "Eliminar citas"
4. new()
5. cargarCitas(UsuarioTutor)
6. execSQL("SELECT * FROM Alumno, CitaAlumnoTutor WHERE Alumno.IdAlumno=CitaAlumnoTutor.IdAlumno AND Colegiado=%UsuarioTutor%")
7. new()
8. next()
9. getInteger("IdAlumno") : Integer
10. getString("Nombre") : String
12. getString("Apellidos") : String
12. getDate("Fecha") : Date
13. getDate("Hora") : Date
14. getString("Motivo") : String
15. close()
16. El tutor elige una cita y pulsa "Eliminar cita"
17. eliminarCita(IdAlumnoSeleccionado, UsuarioTutor, FechaSeleccionada, HoraSeleccionada)
18. execSQL("DELETE FROM CitaAlumnoTutor WHERE IdAlumno=%IdAlumnoSeleccionado% AND Colegiado=%UsuarioTutor% AND Fecha=%FechaSeleccionada% AND Hora=%HoraSeleccionada%")

Ilustración 37 - Diagrama Eliminar Cita



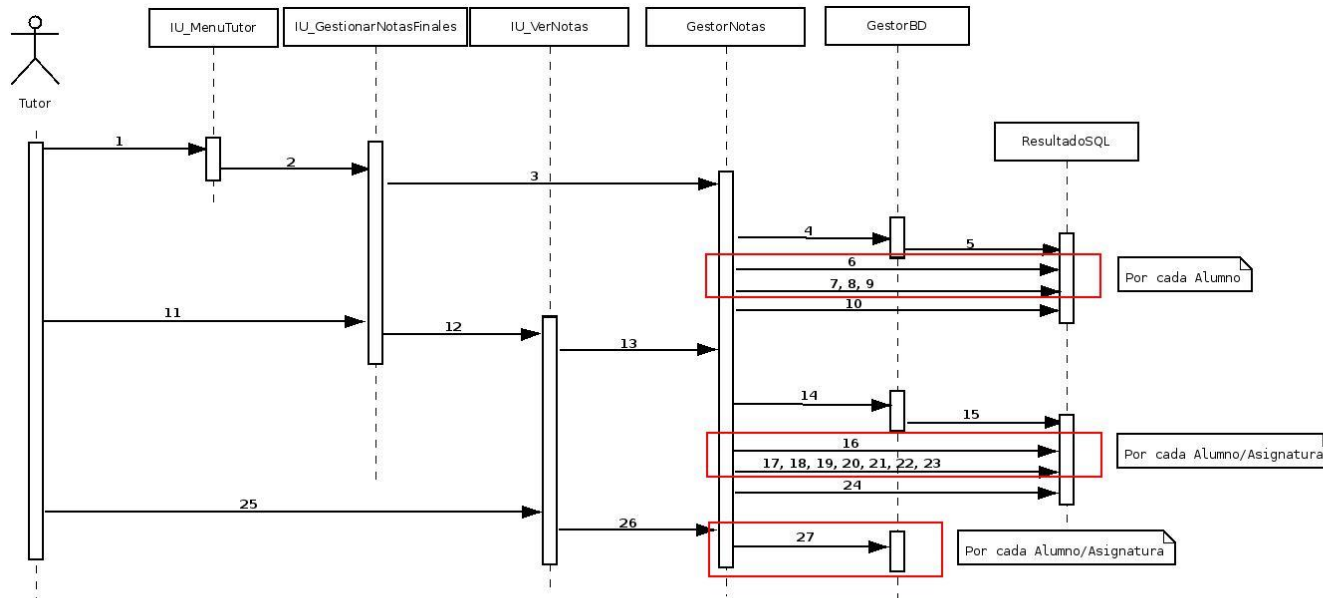
1. El tutor pulsa el botón "Gestionar citas"
2. new()
3. El tutor pulsa el botón "Ver citas"
4. new()
5. cargarCitas(UsuarioTutor)
6. execSQL("SELECT * FROM Alumno, CitaAlumnoTutor WHERE Alumno.IdAlumno=CitaAlumnoTutor.IdAlumno AND Colegiado=%UsuarioTutor or%")
7. new()
8. next()
9. getString("Nombre") : String
10. getString("Apellidos") : String
11. getDate("Fecha") : Date
12. getDate("Hora") : Date
13. getString("Motivo") : String
14. close()

Ilustración 38 - Diagrama Ver Cita



1. El tutor pulsa el botón "Gestionar circulares"
2. new()
3. El tutor pulsa el botón "Enviar circulares"
4. cargarCirculares(): ListaCirculares
5. execSQL("SELECT * FROM Circular")
6. new()
7. next()
8. getInteger("IdCircular"): Integer
9. getString("Texto"): String
10. close()
11. El tutor selecciona una circular y pulsa el botón "Enviar"
12. cargarAlumnosCurso(IdCursoTutor): ListaAlumnos
13. execSQL("SELECT * FROM Alumno WHERE IdCurso=%IdCursoTutor%")
14. new()
15. next()
16. getInteger("IdAlumno"): Integer
 - 16.1. enviarCircular(IdCircular, IdAlumno, Fecha, Hora)
 - 16.2. execSQL("INSERT INTO CirculaEnviada(IdCircular, Colegiado, IdAlumno, Fecha, Hora) VALUES (%IdCircular%, %Tutor%, %IdAlumno%, %Fecha%, %Hora%)")
17. close()

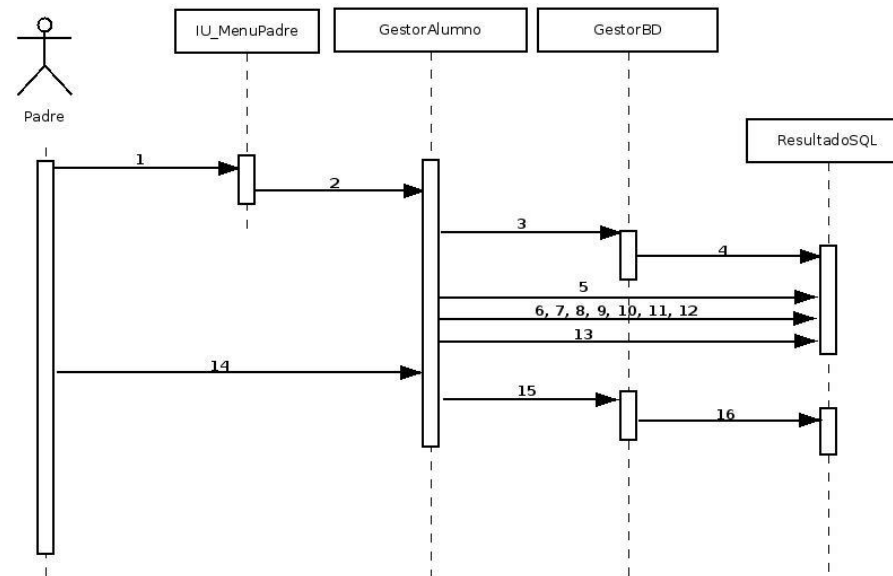
Ilustración 39 - Diagrama Gestionar Circulares



1. El tutor pulsa el botón "Preparar notas finales"
2. new()
3. cargarAlumnos(UsuarioTutor) : ListaAlumnos
4. execSQL("SELECT * FROM Alumno, Tutor WHERE Alumno.IdCurso=Tutor.IdCurso AND Colegiado=%UsuarioTutor%")
5. new()
6. next()
7. getInteger("IdAlumno") : Integer
8. getString("Nombre") : String
9. getString("Apellidos") : String
10. close()
11. El tutor elige un alumno y pulsa "Preparar notas"
12. new()
13. obtenerNotas(IdAlumnoSeleccionado)
14. execSQL("SELECT * FROM NotaAlumnoEnAsignatura, FaltasAlumnoEnAsignatura WHERE NotaAlumnoEnAsignatura.IdAlumno=IdAlumnoSeleccionado AND FaltasAlumnoEnAsignatura.IdAlumno")
15. new()
16. next()
17. getDouble("Nota1") : Double
18. getDouble("Nota2") : Double
19. getDouble("Nota3") : Double
20. getString("Observaciones1") : String
21. getString("Observaciones2") : String
22. getString("Observaciones3") : String
23. getInteger("IdAsignatura") : Integer
24. close()
25. El tutor rellena los comentarios finales y pulsa "Enviar"
26. enviarNotas(IdAlumnoSeleccionado, IdAsignatura, NotaFinal, ObservacionFinal)
27. execSQL("UPDATE NotaAlumnoEnAsignatura SET Final=%NotaFinal%, ObservacionFinal=%ObservacionFinal% WHERE IdAlumno=%IdAlumnoSeleccionado% AND IdAsignatura=%IdAsignatura%")

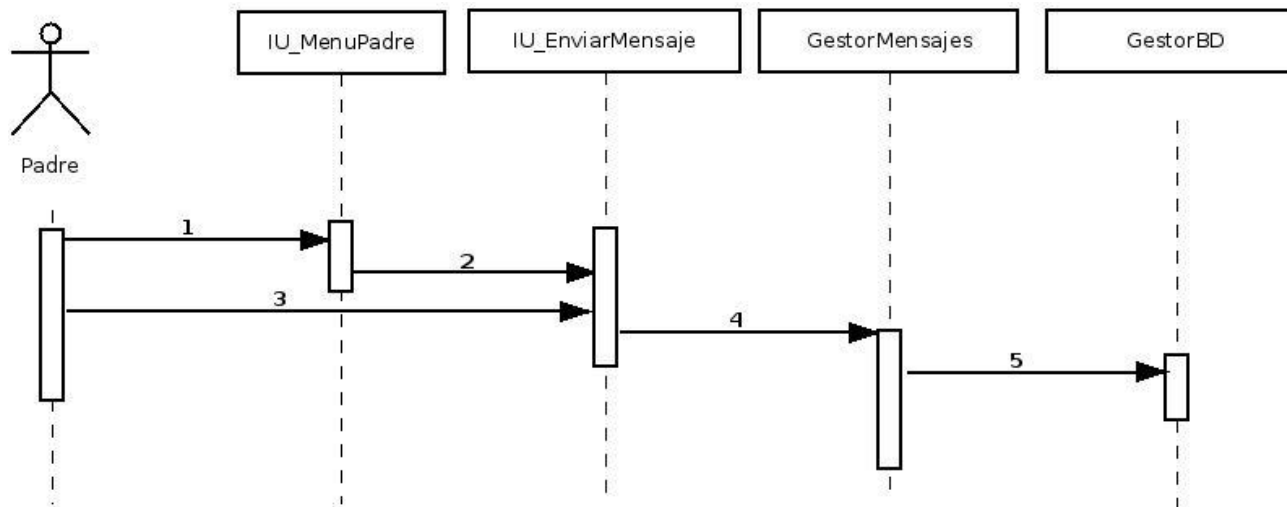
Ilustración 40 - Diagrama Preparar notas

12.5.4 Diagramas de secuencia de Padre



1. El padre pulsa el botón "Ver datos"
2. cargarInfoAlumno(IdAlumno)
3. `execSQL("SELECT * FROM Alumno, Curso WHERE idAlumno=%IdAlumno% AND IdCurso=%IdCurso%")`
4. `new()`
5. `next()`
6. `getString("DNI") : String`
7. `getString("Nombre") : String`
8. `getString("Apellidos") : String`
9. `getString("Teléfono") : String`
10. `getString("FechaNacimiento") : String`
11. `getString("Dirección") : String`
12. `getString("Curso")`
13. `close()`
14. El padre rellena los datos y pulsa "Editar"
15. `guardarInfoAlumno()`
16. `execSQL("UPDATE Alumno SET Teléfono=%Teléfono%, Dirección=%Dirección% WHERE IdAlumno=%IdAlumno%")`

Ilustración 41 - Diagrama Ver Información Hijo



1. El padre pulsa el botón "EnviarMensaje"
2. new()
3. El padre rellna los datos y pulsa el botón "Enviar"
4. new()
5. enviarMensaje(TutorAlumno, Mensaje)
7. execSQL("INSERT INTO MensajeAlumnoProfesor (IdAsignatura, Fecha, Hora, Mensaje) VALUES (%IdTutoria%, %FechaActual%, %HoraActual%, %Mensaje%)")

Ilustración 42 - Diagrama Enviar Mensaje

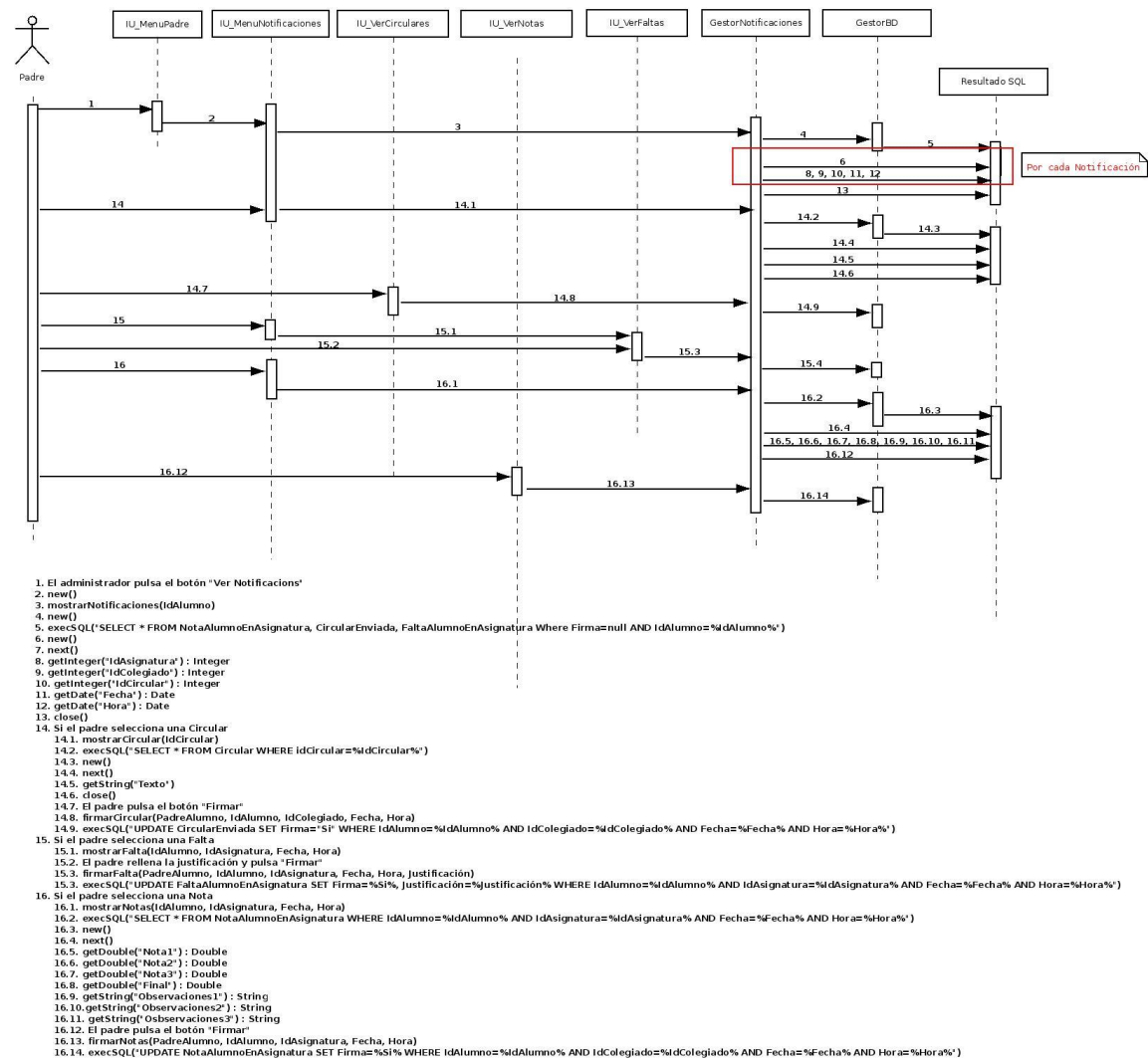
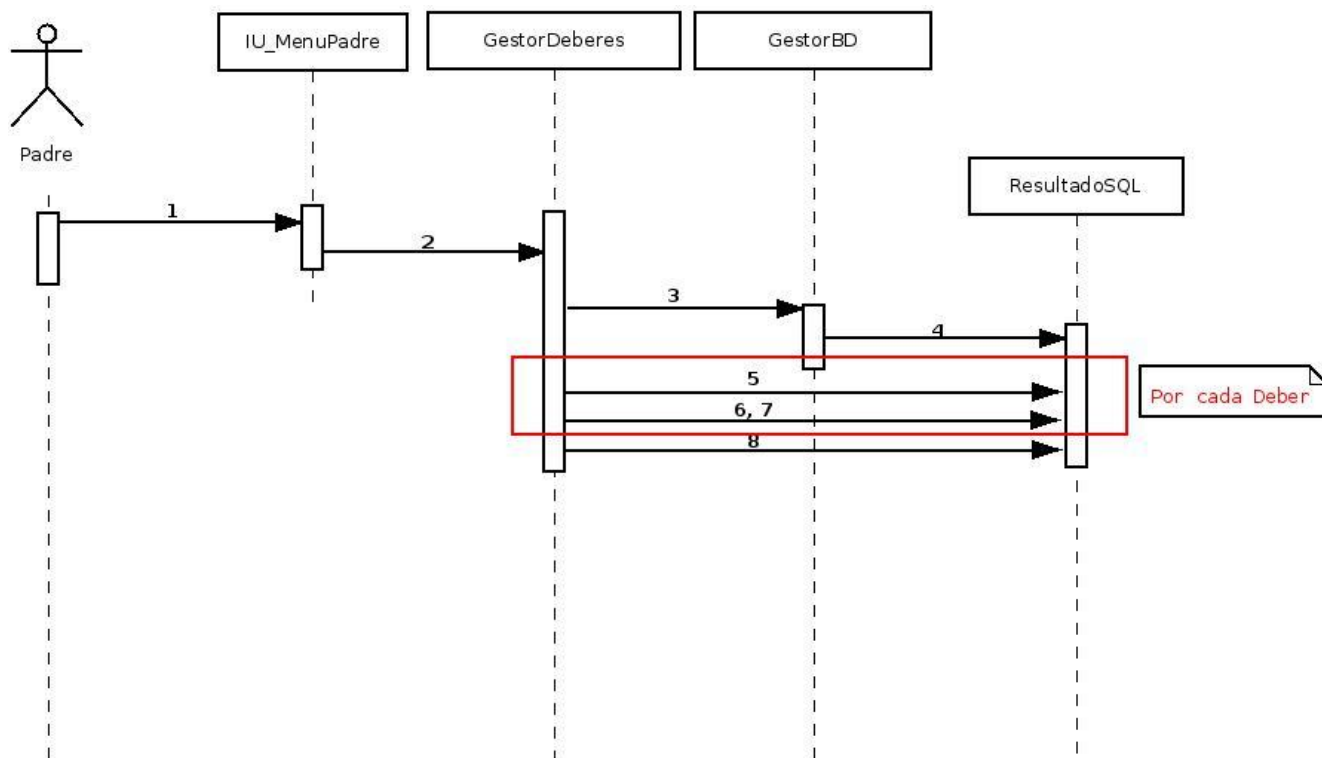
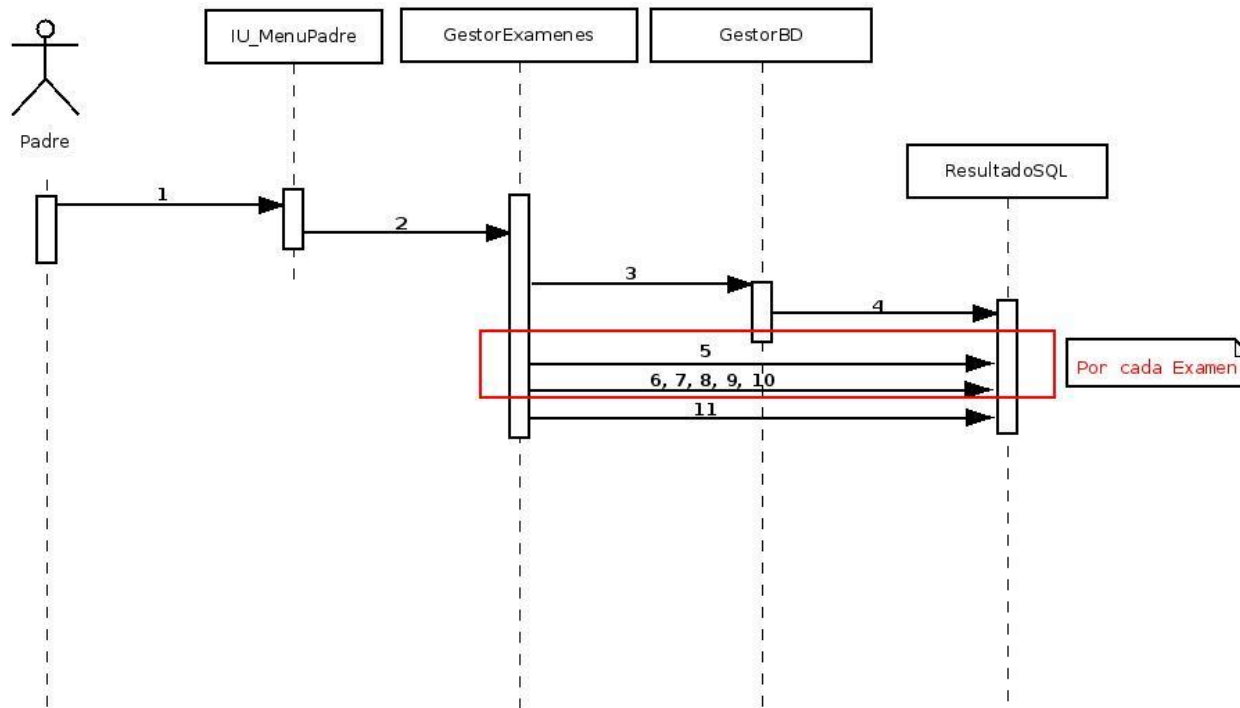


Ilustración 43 - Diagrama Ver Notas



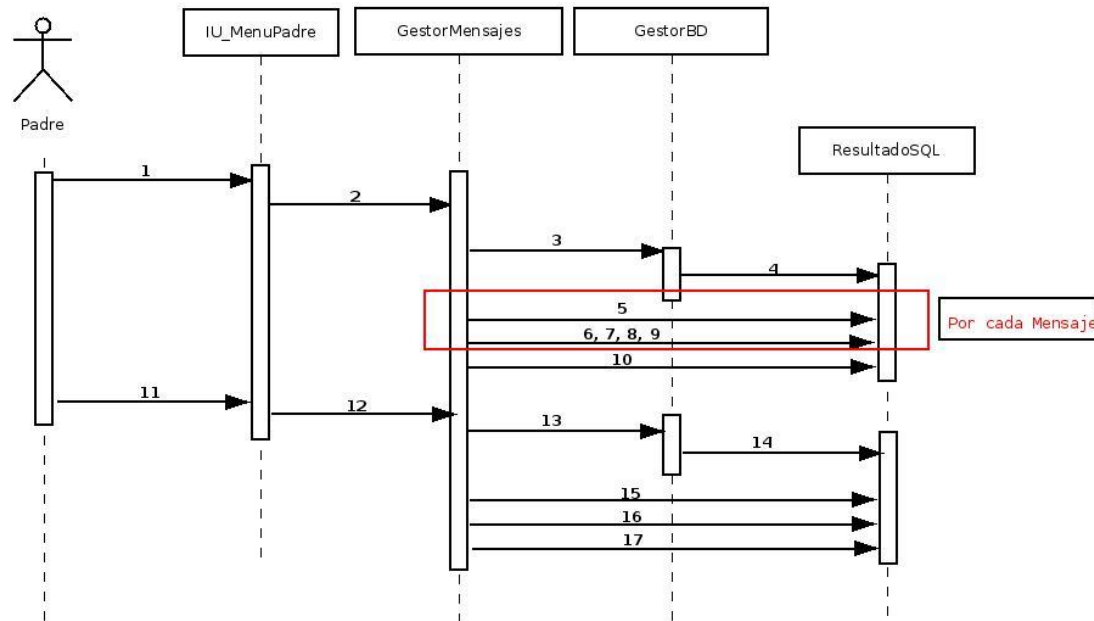
1. El padre pulsa el botón "Ver deberes"
2. cargarDeberes(idAlumno)
3. `execSQL("SELECT * FROM DeberAlumnoEnAsignatura, Deber, Asignatura WHERE idAlumno=%dAlumno% AND IdTarea=%dIdTarea% AND IdAsignatura=%dIdAsignatura%")`
4. `new()`
5. `next()`
6. `getString("Tarea") : String`
7. `getString("Asignatura") : String`
8. `close()`

Ilustración 44 - Diagrama Ver Deberes



1. El padre pulsa el botón "Ver exámenes"
2. cargarExámenes(IdAlumno)
3. `execSQL("SELECT * FROM ExamenAlumnoEnAsignatura, Examen, Asignatura WHERE idAlumno=%IdAlumno% AND IdExamen=%IdExamen% AND IdAsignatura=%IdAsignatura%")`
4. `new()`
5. `next()`
6. `getDate("Fecha") : Date`
7. `getDate("Hora") : Date`
8. `getString("Asignatura") : String`
9. `getDouble("Nota") : Double`
10. `getString("Comentarios")`
11. `close()`

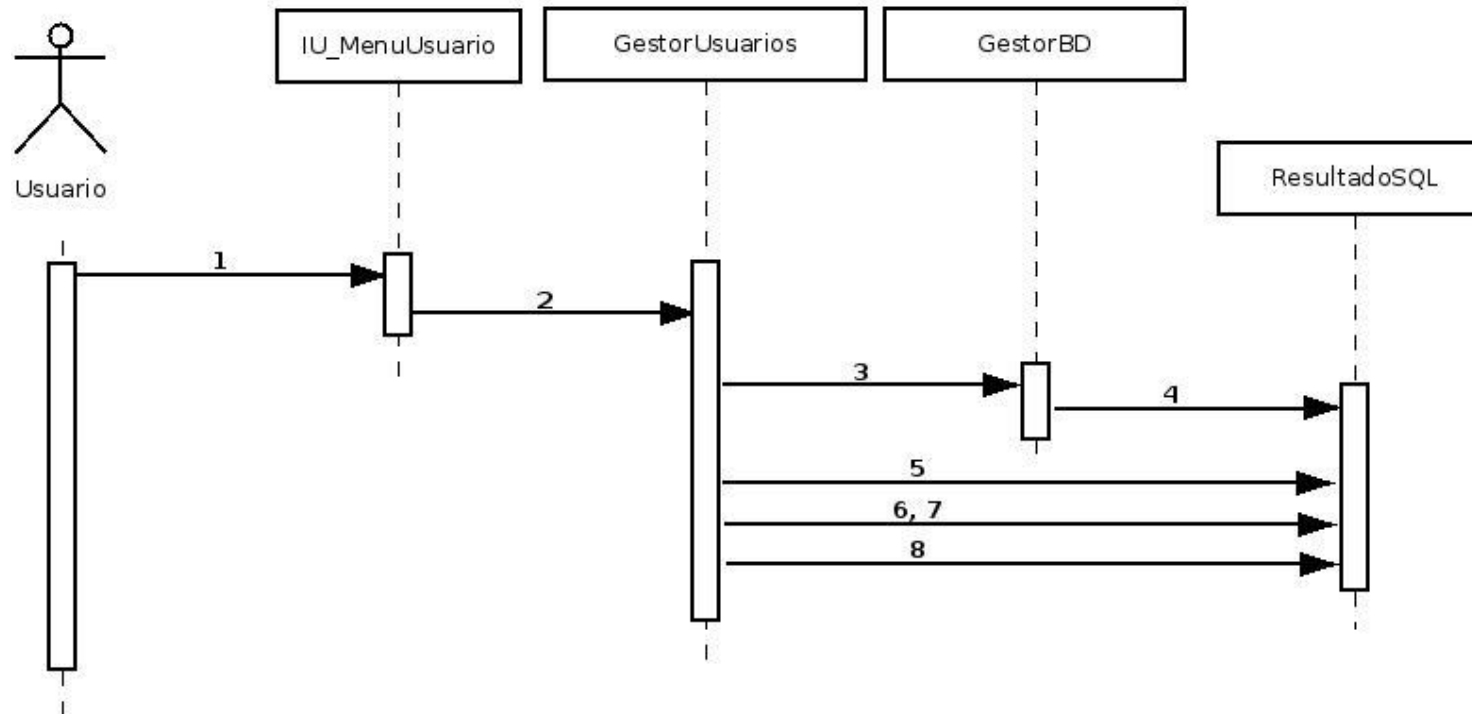
Ilustración 45 - Diagrama Ver Exámenes



1. El padre pulsa el botón "Ver mensajes"
2. cargarMensajes(IdAlumno)
3. `execSQL("SELECT * FROM MensajeAlumnoProfesor, Profesor WHERE idAlumno=%IdAlumno% AND IdColegiado=%IdColegiado%")`
4. `new()`
5. `next()`
6. `getDate("Fecha") : Date`
7. `getDate("Hora") : Date`
8. `getString("Nombre") : String`
9. `getInteger("IdColegiado") : Integer`
10. `close()`
11. El padre selecciona un mensaje y pulsa "Ver mensaje"
12. `cargarMensaje(IdAlumno, IdColegiado, Fecha, Hora)`
13. `execSQL("SELECT * MensajeAlumnoProfesor WHERE IdAlumno=%IdAlumno% AND IdColegiado=%IdColegiado% AND Fecha=%Fecha% AND Hora=%Hora%")`
14. `new()`
15. `next()`
16. `getString("Mensaje") : String`
17. `close()`

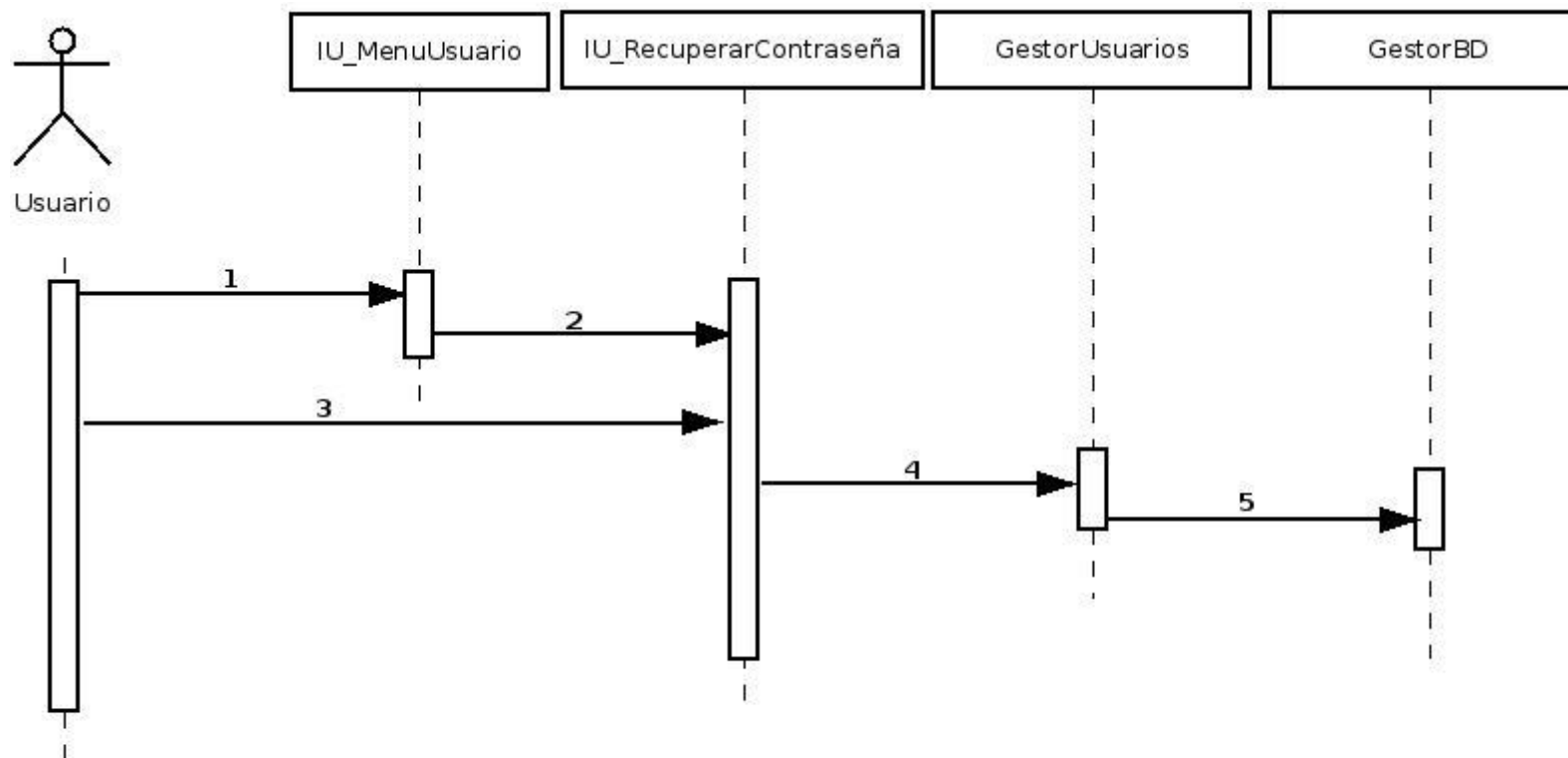
Ilustración 46 - Diagrama Ver Mensajes

12.5.5. Diagramas de secuencia de usuario



1. El usuario rellena su usuario y contraseña y pulsa el botón "Identificarse"
2. `identificarse(usuario, contraseña)`
3. `execSQL("SELECT * FROM Padre WHERE Usuario=%usuario% AND contraseña=%Contraseña%")`
4. `new()`
5. `next()`
6. `getString(" Usuario") : String`
7. `getString(" Contraseña") : String`
8. `close()`

Ilustración 47 - Diagrama Identificarse



1. El usuario rellena su usuario y contraseña y pulsa el botón "Recuperar Contraseña"
2. new()
3. El usuario introduce su número de DNI y pulsa "Recuperar"
4. generarContraseña(usuarioIntroducido)
5. execSQL("UPDATE Padre SET contraseña=%ContraseñaGenerada% WHERE Usuario=%usuarioIntroducido%")

Ilustración 48 - Diagrama Recuperar Contraseña

12.6 Plan de Pruebas

A continuación, se detallan las pruebas realizadas a fondo para cada uno de los actores una vez la implementación está desarrollada, para comprobar las posibles deficiencias que tenga el sistema, ya que a medida que se programaba se iban realizando pruebas simples para comprobar su funcionamiento correcto

12.6.1 Plan de pruebas usuario

Descripción	Acción	Resultado esperado	Resultado real	Correcto	Error	Comentarios
El usuario intenta entrar en la aplicación sin introducir algún o ningún dato	Pulsa el botón "Entrar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El usuario intenta entrar en la aplicación sin introducir correctamente el usuario o la contraseña	Pulsa el botón "Entrar"	Salta error para que rellene el usuario y la contraseña	El sistema funciona según lo esperado	X		-
El usuario intenta entrar en la aplicación introduciendo los datos de manera correcta	Pulsa el botón "Entrar"	Va a la sesión del usuario	El sistema funciona según lo esperado	X		-

El usuario pulsa recuperar contraseña sin introducir el DNI del usuario a recuperar	Pulsa el botón "Recuperar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El usuario introduce un DNI que no está registrado en el sistema	Pulsa el botón "Recuperar"	Salta error para que rellene con un usuario correcto	El sistema funciona según lo esperado	X		-
El usuario intenta recuperar la contraseña de un usuario que existe en el sistema	Pulsa el botón "Recuperar"	La contraseña se modifica correctamente	El sistema funciona según lo esperado, pero no hay manera de notificar al usuario		X	Se envía un correo al usuario con la contraseña generada y se crea funcionalidad para que nada más iniciar sesión solicite un cambio de contraseña
El usuario intenta cambiar la contraseña sin introducir ninguno o todos los datos que se le permiten	Pulsa el botón "Cambiar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El usuario intenta cambiar la contraseña introduciendo una contraseña actual que no corresponde con la que se le ha solicitado	Pulsa el botón "Cambiar"	Salta error para que introduzca la contraseña actual correcta	El sistema funciona según lo esperado	X		-
El usuario intenta cambiar la	Pulsa el botón "Cambiar"	Va a la sesión del usuario	El sistema funciona según lo esperado	X		-

contraseña introduciendo correctamente la contraseña actual						
---	--	--	--	--	--	--

Tabla 69 - Pruebas Usuario

12.6.2 Plan de pruebas administrador

Descripción	Acción	Resultado esperado	Resultado real	Correcto	Error	Comentarios
El administrador intenta crear un profesor/curso/asignatura/alumno/usuario/circular sin introducir alguno o todos los campos que se le permiten	Pulsa el botón "Crear"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El administrador intenta crear un profesor/curso/asignatura/alumno/usuario/circular con todos los datos introducidos y que ya existe en el sistema	Pulsa el botón "Crear"	Saltar error de que ya existe en el sistema	El sistema funciona según lo esperado	X		-
El administrador intenta crear un profesor/curso/asignatura/alumno/usuario/circular con todos los datos introducidos y que no existe en el sistema	Pulsa el botón "Crear"	Se crea correctamente	El sistema funciona según lo esperado	X		-
El administrador intenta editar un profesor/curso/asignatura/alumno/usuario/circular sin seleccionar ninguno	Pulsa el botón "Editar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El administrador intenta editar un profesor/curso/asignatura/alumno/usuario/circular habiendo seleccionado anteriormente uno	Pulsa el botón "Editar"	Se edita correctamente	El sistema funciona según lo esperado	X		-

El administrador intenta eliminar un profesor/curso/asignatura/alumno/usuario/circular sin haber seleccionado ninguno	Pulsa el botón "Eliminar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El administrador intenta eliminar un profesor/curso/asignatura/alumno/usuario/circular habiendo seleccionado anteriormente uno	Pulsa el botón "Eliminar"	Se eliminar correctamente	El sistema funciona según lo esperado	X		-

Tabla 70 - Pruebas Administrador

12.6.3 Plan de pruebas profesor

Descripción	Acción	Resultado esperado	Resultado real	Correcto	Error	Comentarios
El profesor intenta crear un deber/examen sin introducir alguno o todos los campos que se le permiten	Pulsa botón "Crear"	el formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El profesor intenta crear un deber/examen con todos los datos introducidos y que ya existe en el sistema	Pulsa botón "Crear"	el Salta error de que ya existe en el sistema	El sistema funciona según lo esperado	X		-
El profesor intenta crear un deber/examen con todos los datos introducidos y que no existe en el sistema	Pulsa botón "Crear"	el Se crea correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta editar un deber/examen sin seleccionar ningún elemento	Pulsa botón "Editar"	el formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El profesor intenta editar un deber/examen habiendo seleccionado anteriormente uno	Pulsa botón "Editar"	el Se edita correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta eliminar un deber/examen sin haber seleccionado ninguno	Pulsa botón "Eliminar"	el formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El profesor intenta eliminar un deber/examen habiendo seleccionado anteriormente uno	Pulsa botón "Eliminar"	el Se elimina correctamente	El sistema funciona según lo esperado	X		-

El profesor intenta corregir un deber/examen sin haber seleccionado y rellenado alguno o todos los campos que se le permiten	Pulsa el botón "Corregir"	El formulario informa que los campos son requeridos	Marca que se corrige, pero no se ha seleccionado nada		X	Para los desplegables comprobar que está marcado un elemento válido si no saltar error
El profesor intenta corregir un deber/examen sin habiendo rellenado todos los campos que se le permiten	Pulsa el botón "Corregir"	El deber/examen se corrige correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta responder un mensaje sin tener ninguno seleccionado	Pulsa el botón "Responder"	El formulario informa que los campos son requeridos	Marca que se envía, pero no se ha seleccionado nada		X	Para los desplegables comprobar que está marcado un elemento válido si no saltar error
El profesor intenta responder un mensaje habiendo seleccionado uno	Pulsa el botón "Responder"	El mensaje se envía correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta enviar un mensaje sin haber rellenado alguno o todos los campos que se le permite	Pulsa el botón "Enviar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El profesor intenta enviar un mensaje habiendo rellenado todos los campos que se le permite	Pulsa el botón "Enviar"	El mensaje se envía correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta preparar las notas de un alumno es una asignatura sin rellena alguno o ninguno de los campos que se le permite	Pulsa el botón "Preparar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El profesor intenta preparar las notas de un alumno es una asignatura rellenando todos de los campos que se le permite y habiendo seleccionado un	Pulsa el botón "Preparar"	Salta error de que ya han sido preparadas	El sistema funciona según lo esperado	X		-

trimestre en el que las notas finales ya han sido enviadas a los padres						
El profesor intenta preparar las notas de un alumno es una asignatura rellenando todos de los campos que se le permite y habiendo seleccionado un trimestre en el que las notas finales no han sido enviadas a los padres	Pulsa el botón "Preparar"	Las notas se preparan correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta añadir una falta sin haber rellenado alguno o ningún campo de los que se le permite	Pulsa el botón "Crear"	El formulario informa que los campos son requeridos	Marca que ha sido añadida, pero no es así		X	Para los desplegables comprobar que está marcado un elemento válido si no saltar error
El profesor intenta añadir una falta habiendo rellenado todos los campos que se le permite	Pulsa el botón "Crear"	La falta se crea correctamente	El sistema funciona según lo esperado	X		-
El profesor intenta eliminar una falta sin haber seleccionado ninguna	Pulsa el botón "Eliminar"	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El profesor intenta eliminar una falta habiendo seleccionado alguna	Pulsa el botón "Eliminar"	La falta se elimina correctamente	El sistema funciona según lo esperado	X		-

Tabla 71 - Pruebas profesor

12.6.4 Plan de pruebas tutor

Descripción	Acción	Resultado esperado	Resultado real	Correcto	Error	Comentarios
El tutor intenta enviar una circular sin haber seleccionado ninguna	Pulsa botón "Enviar"	el formulario informa que los campos son requeridos	Marca que se envía, pero no se ha seleccionado nada		X	Para los desplegables comprobar que está marcado un elemento válido si no saltar error
El tutor intenta enviar una circular habiendo seleccionado una	Pulsa botón "Enviar"	el La circular se envía correctamente	El sistema funciona según lo esperado	X		-
El tutor intenta añadir una hora disponible para cita sin añadir alguno o ninguno los campos que se le permite	Pulsa botón "Crear"	el formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El tutor intenta añadir una hora disponible para cita añadiendo todos los campos que se le permite y la hora ya existe	Pulsa botón "Crear"	el Salta error de que ya existe	El sistema funciona según lo esperado	X		-
El tutor intenta añadir una hora disponible para cita añadiendo todos los campos que se le permite y la hora no existe	Pulsa botón "Crear"	el La hora se añade correctamente	El sistema funciona según lo esperado	X		-
El tutor intenta eliminar una hora sin haber seleccionado ninguna	Pulsa botón "Eliminar"	el formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El tutor intenta eliminar una hora habiendo seleccionado una	Pulsa botón "Eliminar"	el La hora se elimina correctamente	El sistema funciona según lo esperado	X		-

El tutor intenta enviar las notas finales de un trimestre sin haber rellenado alguno o ninguno de los campos que se le permite	Pulsa botón "Enviar"	el	El formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El tutor intenta enviar las notas finales de un trimestre habiendo rellenado todos los campos que se le permite cuando esas notas ya han sido enviadas	Pulsa botón "Enviar"	el	El formulario informa que los campos son requeridos	Marca que se envía, pero no se ha seleccionado nada		X	Para los desplegados comprobar que está marcado un elemento válido si no saltar error
El tutor intenta enviar las notas finales de un trimestre habiendo rellenado todos los campos que se le permite cuando esas notas no han sido enviadas	Pulsa botón "Enviar"	el	Las notas se envían correctamente	El sistema funciona según lo esperado	X		-

Tabla 72 - Pruebas Tutor

12.6.5 Plan de pruebas padre

Descripción	Acción	Resultado esperado	Resultado real	Correcto	Error	Comentarios
El padre intenta editar la información de su hijo sin haber seleccionado ninguno	Pulsa botón "Editar"	el formulario informa que los campos son requeridos	El sistema funciona según lo esperado	X		-
El padre intenta editar la información de su hijo habiendo seleccionado uno	Pulsa botón "Editar"	el La información se edita correctamente	El sistema funciona según lo esperado	X		-
El padre intenta firmar las notas finales/circulares sin haber seleccionado alguno o ninguno de los datos que se le permite	Pulsa botón "Firmar"	el formulario informa que los campos son requeridos	Marca que se firman, pero no se ha seleccionado nada		X	Para los desplegables comprobar que está marcado un elemento válido si no saltar error
El padre intenta firmar las notas finales/circulares habiendo seleccionado todos los datos que se le permite, pero con un certificado no válido	Pulsa botón "Firmar"	el Salta error de que no es válido	El sistema funciona según lo esperado	X		-
El padre intenta firmar las notas finales/circulares habiendo seleccionado todos los datos que se le permite, pero con un certificado válido y las notas firmadas anteriormente	Pulsa botón "Firmar"	el Salta error de que ya han sido firmadas	El sistema funciona según lo esperado	X		-
El padre intenta firmar las notas finales/circulares habiendo seleccionado todos los datos que se le permite, pero con un certificado válido y las notas sin firmar anteriormente	Pulsa botón "Firmar"	el Las notas se firman correctamente	El sistema funciona según lo esperado	X		-
El padre intenta reservar una hora sin haber seleccionado ninguna	Pulsa botón "Reservar"	el formulario informa que los	El sistema funciona según lo esperado	X		-

		campos son requeridos				
El padre intenta reservar una hora habiendo seleccionado una	Pulsa el botón "Reservar"	La hora se reserva correctamente	El sistema funciona según lo esperado	X		-

Tabla 73 - Pruebas Padre

13. ESFUERZO PLANIFICADO VS. ESFUERZO REAL

En la siguiente tabla se detallan las desviaciones de plazos de cada tarea, se puede observar las horas planificadas inicialmente, las horas dedicadas realmente y la desviación producida para cada una de ellas.

FASE/TAREA	HORAS PLANIFICADAS	HORAS REALES	DESVIACIÓN
Especificación de necesidades	41	39	-5%
Idea del proyecto	10	6	-40%
Aplicaciones similares	2	2	0%
Entrevista con la directora del proyecto	2	1	-50%
Enunciar objetivos	2	3	50%
Proyecto de viabilidad	25	27	8%
Análisis	47	43	-9%
Reunión con la directora del proyecto	2	1	-50%
Captura de requisitos	7	10	43%
Búsqueda sobre lenguajes y herramientas	25	22	-12%
Selección de herramientas	3	2	-33%
Documentación	10	8	-20%
Diseño	47	44	-6%
Diseño de la aplicación	35	31	-11%
Documentación	12	13	8%
Aprendizaje	55	67	22%
Estudio del lenguaje y herramientas	40	45	13%
Aprendizaje de la tecnología	15	22	47%
Implementación	239	305	40%
Implementación de la interfaz gráfica	30	27	-10%
Implementación de la base de datos	20	18	-10%
Implementación del código	170	230	35%
Integración del sistema	12	20	67%
Documentación	7	10	43%
Pruebas	36	50	39%
Realización de pruebas	8	12	50%
Corrección de errores	16	14	-13%
Mejora del funcionamiento y eficiencia	7	10	43%
Documentación	5	4	-20%
Documentación	12	15	25%
Manual de usuario	5	5	0%
Memoria del proyecto	7	10	43%
Cierre	2	2	0%

Empaquetado documentos y entrega	2	2	0%
Trabajo Fin de Grado	533	565	6%

Tabla 74 - Esfuerzo planificado vs esfuerzo real

A continuación, mostramos gráficamente la diferencia entre la planificación inicial y el esfuerzo real, como puede observarse, la mayor desviación de plazos de encuentra en la parte de desarrollo, ya que sobre todo la implementación de la firma electrónica fue de lo más costoso del proyecto debido a la falta de conocimientos e información.

	Necesidades	Análisis	Diseño	Aprendizaje	Implementación	Pruebas	Documentación	Cierre
Nº horas planificadas	41	47	47	55	239	36	12	2
Nº horas reales	39	43	44	67	305	50	15	2

Tabla 75 - Comparación horas planificadas vs reales

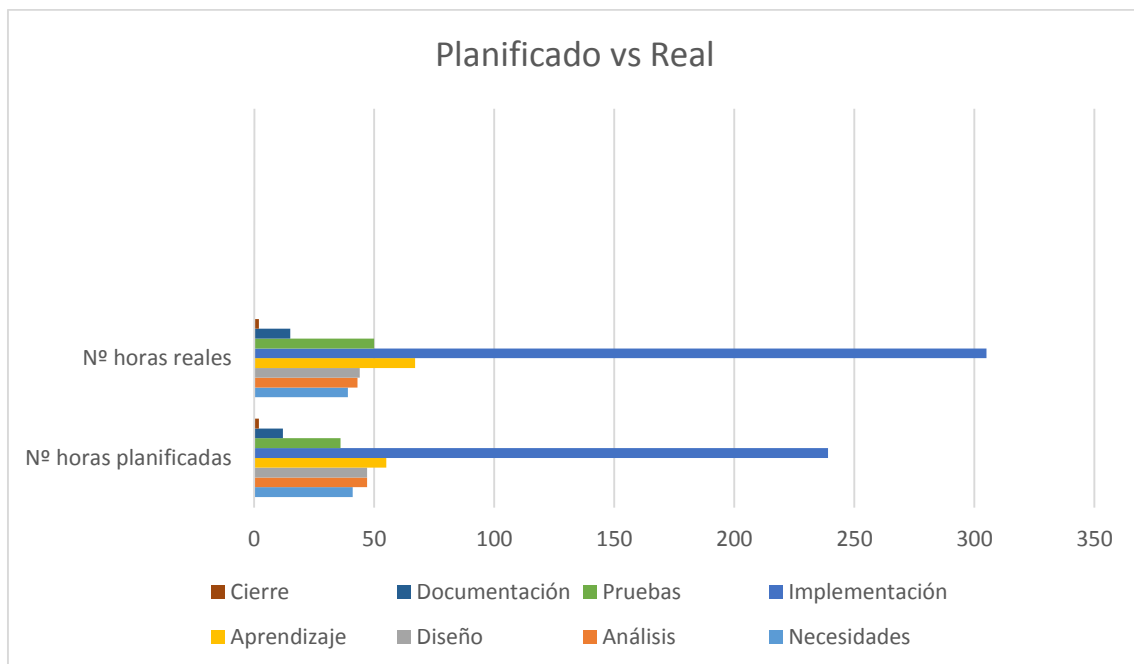


Ilustración 49 - Gráfica horas planificadas vs horas reales

14. POSIBLES MEJORAS

Tras el desarrollo de este proyecto, hemos podido detectar mejoras que podría hacerse en un futuro para una posible implantación en algún centro que lo desee o que no se han tenido en cuenta en el diseño inicial, pero podrían mejorar nuestro sistema:

- Realizar un sistema en **varios idiomas**: se podría plantear la necesidad de desarrollarlo en castellano, euskera e inglés.
- Invertir en un **certificado SSL**: en el momento en que este sistema vaya a ser implantado en algún centro, se debería invertir en un certificado de confianza que nos permita la navegación https de manera segura y certifique el sitio de confianza. Esto revalorizará nuestro sistema de cara a la seguridad.

15. CONCLUSIONES

A pesar de que al proyecto se le han dedicado más horas de las planificadas en un momento inicial, el sistema cumple con sus objetivos establecidos en un primer momento. Se ha creado la base de una aplicación que podría dar soporte a diferentes centros educativos, dotándolo de la seguridad inicial planteada.

A nivel personal, a pesar de tener una dedicación constante en investigación y aprendizaje de nuevas herramientas, el desarrollo ha cumplido sus expectativas. Se han adquirido nuevos conocimientos y además a diferencia de lo que se realiza durante la carrera, se ha conseguido sacar adelante un proyecto desde cero, abarcando de manera individual todas las fases del mismo cumpliendo en la medida de lo posible de los tiempos estipulados.

Como previsión de futuro, la idea no surgió con intención de llegar a ser implantada en algún colegio, si no que fue por la preferencia de desarrollos dedicados a la gestión lo que me impulsó a abordarlo, sin embargo, una vez completado el proyecto, este sistema con más dedicación y perfilándolo a las necesidades de los clientes, podría dar ser un complemento perfecto a los sistemas educativos de hoy en día.

16. BIBLIOGRAFÍA

Agencia Española de Protección de Datos -

<https://www.agpd.es/portalwebAGPD/index-ides-idphp.php>

Portal del DNI electrónico, Cuerpo Nacional de Policía -

<http://www.dnielectronico.es/PortalDNIe/>

HTML5 Tutorial - http://www.w3schools.com/html/html5_intro.asp

PHP Manual - <https://secure.php.net/manual/es/index.php>

Certificados SSL - <http://www.genbeta.com/seguridad/que-es-un-certificado-ssl-y-por-que-deberia-importarte>