



Universidad de Valladolid

**ESCUELA DE INGENIERÍA INFORMÁTICA
(SEGOVIA)**

**Grado en Ingeniería Informática de Servicios y
Aplicaciones**

EFAIDia.

Alumno: Clara María Herranz de Andrés.

Tutores: José Vicente Álvarez Bravo.

Aníbal Bregón Bregón.

Resumen

EFAIDia es una aplicación cuya misión es complementar en tiempo real la labor del docente de Educación Física, para la mejora de la evaluación. A través de esta aplicación, con la introducción de datos (escritos y transcritos a partir de voz) el docente podrá realizar la toma de anotaciones en tiempo real, su consulta y la planificación temporal de las clases, dividiéndolas en objetivos que permitirán la evaluación diaria.

Este es el principal motivo por el cual se ha desarrollado una aplicación móvil que permita la organización y planificación de las clases y la evaluación individual de cada alumno.

En conclusión, EFAIDia busca facilitar la labor docente en el área de Educación Física. Actuando de medio con el que el docente podrá planificar sus clases, redactar y consultar las fichas de sus alumnos, con un gran ahorro de tiempo, complementando la enseñanza tradicional de seguimiento y evaluación que se emplean en esta materia.

Palabras clave: evaluación, ahorro de tiempo, Educación Física, alumnos, anotaciones, tiempo real.

Índice de contenidos

Índice de tablas	7
Índice de figuras	9
Capítulo 1 - Introducción.....	13
1.1. Motivación	14
1.2. Objetivos	15
1.3. Alcance del proyecto.....	15
1.4. Reglas de negocio.....	17
1.5. Organización del documento.....	17
1.6. Contenido del CD-ROM	18
Capítulo 2 - Estado del arte.....	19
2.1. Conclusiones del estado del arte	23
Capítulo 3 - Gestión del proyecto	27
3.1. Metodología	27
3.2. Fases de trabajo y estimación temporal	28
3.3. Estimaciones.....	29
3.3.1. Estimación de costes por Puntos de Función (PF).....	30
3.3.2. Estimación de costes por COCOMO	34
3.4. Presupuestos.....	36
3.4.1. Presupuesto inicial	36
3.4.2. Presupuesto final.....	37
3.5. Conclusiones.....	39
Capítulo 4 - Análisis del sistema.....	41
4.1. Actores del sistema	41
4.2. Requisitos de usuario	42
4.2.1. Diagrama de casos de uso	44
4.2.2. Especificación de casos de uso.....	45
4.3. Requisitos no funcionales	66
4.4. Requisitos de información.....	66
4.5. Diseño conceptual.....	67
Capítulo 5 - Diseño de Software.....	69
5.1. Arquitectura lógica	69
5.1.1. Arquitectura lógica del sistema.....	69
5.1.2. Arquitectura lógica de la aplicación	71
5.2. Arquitectura física	73

5.3.	Diagrama de clases.....	74
5.3.1.	Paquete SQLite.....	75
5.3.2.	Paquete modelo.....	76
5.3.3.	Paquete activity.....	77
5.3.4.	Paquete util.....	78
5.3.5.	Paquete listener.....	79
5.4.	Diagramas de secuencia.....	79
5.5.	Modelo lógico.....	85
5.6.	Diccionario de datos.....	86
5.7.	Diseño de la interfaz.....	88
Capítulo 6 -	Implementación.....	105
6.1.	Tecnologías utilizadas.....	105
6.2.	Herramientas utilizadas.....	108
6.3.	Estructura del proyecto.....	109
Capítulo 7 -	Pruebas.....	111
7.1.	Pruebas de caja negra.....	111
Capítulo 8 -	Manuales.....	119
8.1.	Manual de instalación.....	119
8.2.	Manual de usuario.....	119
Capítulo 9 -	Conclusiones y líneas futuras.....	133
9.1.	Conclusiones.....	133
9.2.	Líneas futuras.....	134
Capítulo 10 -	Bibliografía.....	135

Índice de tablas

Tabla 1.- Comparación de características de las aplicaciones.	24
Tabla 2.- Estimación temporal de las fases del desarrollo.	29
Tabla 3.- Pesos de los dominios de información según complejidad.	30
Tabla 4.- Estimación de las líneas de código por lenguaje de programación.	31
Tabla 5.- Número de entradas.	31
Tabla 6.- Número de salidas.	31
Tabla 7.- Número de consultas externas.....	32
Tabla 8.- Número de ficheros lógicos internos.	32
Tabla 9.- Número de ficheros lógicos externos.....	32
Tabla 10.- Cálculo de puntos de función no ajustados.	32
Tabla 11.- Cálculo del factor de ajuste.	33
Tabla 12.- Constantes para cada modelo COCOMO.	34
Tabla 13.- Factores de coste del esfuerzo.	34
Tabla 14.- Presupuesto Hardware.....	36
Tabla 15.- Presupuesto Software.	36
Tabla 16.- Presupuesto de desarrollo.	37
Tabla 17.- Presupuesto total.	37
Tabla 18.- Presupuesto Hardware.....	38
Tabla 19.- Presupuesto Software.	38
Tabla 20.- Presupuesto de desarrollo.	38
Tabla 21.- Presupuesto Final.	39
Tabla 22.- Usuario genérico.	41
Tabla 23.- CU-01: Registro.....	45
Tabla 24.- CU-02: Login.	46
Tabla 25.- CU-03: Cierre de Sesión.....	46
Tabla 26.- CU-04: Recuperación contraseña.....	47
Tabla 27.- CU-05: Configuración horario.....	47
Tabla 28.- CU-06: Visualizar calendario.....	48
Tabla 29.- CU-07: Crear alarma.	48
Tabla 30.- CU-08: Visualizar alarma.	49
Tabla 31.- CU-09: Activar/Desactivar alarma.	49
Tabla 32.- CU-10: Eliminar alarma.....	50
Tabla 33.- CU-11: Crear clase.	51
Tabla 34.- CU-12: Importar clase.....	52
Tabla 35.- CU-13: Visualizar clase.....	53
Tabla 36.- CU-14: Editar clase.	54
Tabla 37.- CU-15: Crear alumno.	55
Tabla 38.- CU-16: Importar alumno.	55
Tabla 39.- CU-17: Visualizar alumno.	56
Tabla 40.- CU-18: Editar alumno.	57
Tabla 41.- CU-19: Eliminar alumno.	58
Tabla 42.- CU-20: Crear anotación.	59

Tabla 43.- CU-21: Editar anotación.	60
Tabla 44.- CU-22: Eliminar anotación.	60
Tabla 45.- CU-23: Visualizar horario.	61
Tabla 46.- CU-24: Crear objetivo.	61
Tabla 47.- CU-25: Importar objetivo.	62
Tabla 48.- CU-26: Asignar objetivo.	63
Tabla 49.- CU-27: Aprobar objetivo.	64
Tabla 50.- CU-28: Eliminar objetivo.	64
Tabla 51.- CU-29: Borrar datos.	65
Tabla 52.- Tabla alumno.	87
Tabla 53.- Tabla objetivo.	87
Tabla 54.- Tabla anotación.	87
Tabla 55.- Tabla clase.	87
Tabla 56.- Tabla horario.	87
Tabla 57.- Tabla objetivoalumno.	88
Tabla 58.- Tabla alarma.	88
Tabla 59.- Diseño pantalla "Inicio de Sesión (Login)".	89
Tabla 60.- Diseño pantalla "Registro de Usuario".	90
Tabla 61.- Diseño pantalla "Horario (Pantalla principal)".	91
Tabla 62.- Diseño pantalla "Calendario".	92
Tabla 63.- Diseño pantalla "Listado Clases".	93
Tabla 64.- Diseño pantalla "Clase".	94
Tabla 65.- Diseño pantalla "Ficha de Alumno".	95
Tabla 66.- Diseño pantalla "Listado de Alarmas".	96
Tabla 67.- Diseño pantalla "Creación de Alarma".	97
Tabla 68.- Diseño pantalla "Creación de Objetivo".	98
Tabla 69.- Diseño pantalla "Creación de Anotaciones".	98
Tabla 70.- PCN Identificación de usuario.	111
Tabla 71.- PCN Cerrar sesión.	112
Tabla 72.- PCN Creación de alarma.	112
Tabla 73.- PCN Borrar todo.	112
Tabla 74.- PCN Visualizar ficha alumno.	112
Tabla 75.- PCN Asociación de objetivo.	113
Tabla 76.- PCN Visualizar alumnos.	113
Tabla 77.- PCN Registrar usuario.	113
Tabla 78.- PCN Importar clases.	113
Tabla 79.- PCN Editar foto del alumno.	114
Tabla 80.- PCN Exportar datos.	114
Tabla 81.- Lanzar evento alarma.	114
Tabla 82.- PCN Desactivar alarma.	115

Índice de figuras

Figura 1.- Diagrama de características.....	16
Figura 2.- Interfaz de “Class Dojo”.....	20
Figura 3.- Interfaz de “GoConqr”.....	21
Figura 4.- Interfaz de “Teacher Aide Pro”.....	22
Figura 5.- Interfaz de “Cuaderno del profesor”.....	23
Figura 6.- Diagrama de Gantt.....	29
Figura 7.- Diagrama de Casos de Uso.....	44
Figura 8.- Modelo Entidad-Relación BBDD Clase.....	67
Figura 9.- Modelo Entidad-Relación BBDD Alarma.....	68
Figura 10.- Arquitectura lógica del sistema.....	70
Figura 11.- Arquitectura lógica de la aplicación.....	71
Figura 12.- Arquitectura física.....	73
Figura 13.- Diagrama de Clases, representación de paquetes.....	74
Figura 14.- Diagrama de Clases, representación del paquete SQLite.....	75
Figura 15.- Diagrama de Clases, representación del paquete modelo.....	76
Figura 16.- Diagrama de Clases, representación del paquete activity.....	77
Figura 17.- Diagrama de Clases, Representación del paquete util.....	78
Figura 18.- Diagrama de Clases, representación del paquete listener.....	79
Figura 19.- Diagrama de secuencia - Registro en el sistema.....	79
Figura 20.- Diagrama de secuencia - Login en el sistema.....	80
Figura 21.- Diagrama de secuencia - Recuperar contraseña.....	80
Figura 22.- Diagrama de secuencia - Crear Clase.....	81
Figura 23.- Diagrama de secuencia - Crear anotación.....	82
Figura 24.- Diagrama de secuencia - Importar Objetivo.....	82
Figura 25.- Diagrama de secuencia - Eliminar objetivo.....	83
Figura 26.- Diagrama de secuencia - Editar alumno.....	83
Figura 27.- Diagrama de secuencia - Exportar datos.....	84
Figura 28.- Diagrama de secuencia - Eliminar todo.....	84
Figura 29.- Diagrama de secuencia - Cerrar sesión.....	85
Figura 30.- Modelo Lógico, base de datos Clase.....	85
Figura 31.- Modelo Lógico, base de datos Alarma.....	86
Figura 32.- Registro y Login.....	99
Figura 33.- Configuración de horario y Horario.....	99
Figura 34.- Calendario y Menú.....	100
Figura 35.- Creación de alarmas y Listado de Alarmas.....	100
Figura 36.- Crear Clase y Listado de Clases.....	101
Figura 37.- Crear Alumno y Clase.....	101
Figura 38.- Creación de Objetivo y su asociación.....	102
Figura 39.- Ficha Alumno.....	102
Figura 40.- Creación de anotaciones.....	103
Figura 41.- Captura de registro.....	120
Figura 42.- Captura de Login.....	120

Figura 43.- Captura configuración horario.....	121
Figura 44.- Captura de Pantalla principal y Menú.....	122
Figura 45.- Fichero importación clases.	122
Figura 46.- Fichero importación Alumnos.....	123
Figura 47.- Fichero importación Objetivos.....	123
Figura 48.- Capturas de creación de objetivos, clases y alumnos.....	124
Figura 49.- Capturas de seguimiento de una clase.	125
Figura 50.- Capturas de la pantalla ficha alumno.....	126
Figura 51.- Captura de pantalla de asociación de objetivo.	127
Figura 52.- Captura de pantalla del calendario.....	128
Figura 53.- Captura de pantalla de representación y lista de alarmas.	129

SECCIÓN 1 – INTRODUCCIÓN

Capítulo 1 - Introducción

Cada vez es mayor la influencia que tienen las TIC (Tecnologías de la información y la comunicación) en nuestra vida diaria, ayudándonos tanto en las tareas cotidianas como en nuestro entorno profesional.

El primer Smartphone fue creado por IBM en 1994, denominado “IBM Simon Personal Communicator”. Viendo sus especificaciones, dista mucho de lo que hoy en día se considera un Smartphone. No cosechó gran éxito, por lo que apenas duró un año a la venta. Fue la presentación del primer Iphone, creado por Apple en 2007, lo que revolucionó el concepto e impulsó su auge entre la población. A partir de ese momento el uso de Internet se instauró de manera generalizada en los dispositivos móviles.

El nacimiento de esta tecnología ha cambiado las costumbres de la mayor parte de la población, con un acceso a Internet acelerado por la implantación de nuevos estándares/tecnologías de comunicación, que aumentaron considerablemente el ancho de banda, proporcionando una gran cantidad de información accesible desde cualquier lugar.

A su vez, el uso de Smartphones facilita la realización de tareas con respecto a otros dispositivos que se usaban con anterioridad, gracias a la incorporación de componentes como reloj, cámara o GPS. La unificación y reducción de estos elementos ha favorecido la aparición de múltiples aplicaciones móviles, con utilidades tan diversas como el entretenimiento (videojuegos, televisión), la mejora en nuestra vida cotidiana (compras online) y laboral (organizadores temporales).

En cuanto a su presencia en el ámbito laboral, comienza a tener un uso casi obligatorio, haciendo al empleado más productivo y agilizando su actividad en casos en los que requiere estar localizable, tomar registros, anotaciones, escanear documentos mediante la cámara, y otra gran multitud de tareas.

A su vez, podemos encontrar su implantación en labores educativas, con dos ramas claramente diferenciadas. Por un lado encontraríamos herramientas orientadas a la docencia, con funcionalidades que apoyan el mejor desarrollo del alumno, ampliando la comunicación profesor-alumno, compartiendo recursos (apuntes, tareas) y facilitando la resolución de dudas y conflictos. Por otro lado, tendríamos herramientas enfocadas a la organización escolar, que ayudarán al centro en tareas como la gestión del AMPA, el control de asistencia y la comunicación con los padres.

La incorporación de los dispositivos móviles en las aulas ha favorecido la personalización del aprendizaje, respetando el ritmo de cada alumno y sus necesidades específicas. Se han obtenido evidencias de que la utilización de estas herramientas tiene un impacto positivo en el aprendizaje del alumno, aumentando su motivación, atención y actitud.

Cabe destacar que estas tecnologías no sustituyen a la educación tradicional, sino que la complementan, facilitando al docente la impartición de la materia y su evaluación.

1.1. Motivación

La evaluación en el ámbito de la educación física es compleja, ya que la valoración de los alumnos se realiza mediante la observación de su actividad y ésta requiere de múltiples anotaciones. Al ejecutarse en un entorno abierto, en el que el docente se encuentra en continuo movimiento, se complica la recogida de información necesaria para la posterior calificación.

En esta materia se hace necesario complementar la educación tradicional con nuevas tecnologías, que permitan al docente seguir con su actividad mientras toma anotaciones relativas al comportamiento y la capacidad física del alumno. Para agilizar la recogida de datos sin interrumpir las sesiones, se requiere el desarrollo de un sistema portable, que el docente pueda llevar encima sin que suponga una inconveniencia.

Por estos motivos, se considera necesario el desarrollo de una aplicación móvil que permita al docente visualizar su actividad semanal y tomar rápidas anotaciones que le ayudarán, junto con los criterios de evaluación, a calificar al alumnado.

1.2. Objetivos

El objetivo principal de la aplicación móvil que se va a desarrollar es facilitar a los docentes de la materia de educación física la gestión de sus clases. Este objetivo principal se descompone en los siguientes subobjetivos:

- Facilitar la proyección y planificación de las actividades a realizar en clase mediante el registro de objetivos a cumplir por los alumnos.
- Permitir, en tiempo real, el registro de anotaciones relativas a los alumnos y las actividades realizadas.
- Simplificar la evaluación del alumnado permitiendo mantener historiales personalizados de los datos de cada alumno, mediante el registro del cumplimiento de los objetivos y la creación de anotaciones.
- Simplificar la organización docente mediante la creación de horarios y calendarios que permitan el alta de alarmas/recordatorios.

1.3. Alcance del proyecto

EFAIDia es una aplicación móvil que permitirá realizar la organización de las clases, mostrando los alumnos que pertenecen a cada una de ellas y dando la posibilidad de crear alarmas y comentarios sobre las sesiones. Esta aplicación está diseñada para docentes de la materia de educación física por lo que se centra en la recapitulación de datos sobre los alumnos con la finalidad de poder disponer de más recursos para llevar a cabo su evaluación completa.

Mediante esta aplicación móvil, el docente, previamente registrado e identificado, puede generar su propio horario, alarmas, clases y alumnos, a los que posteriormente, podrá asociar objetivos y anotaciones.

A continuación, se presenta el diagrama en el que se detallan las características que definen la aplicación:

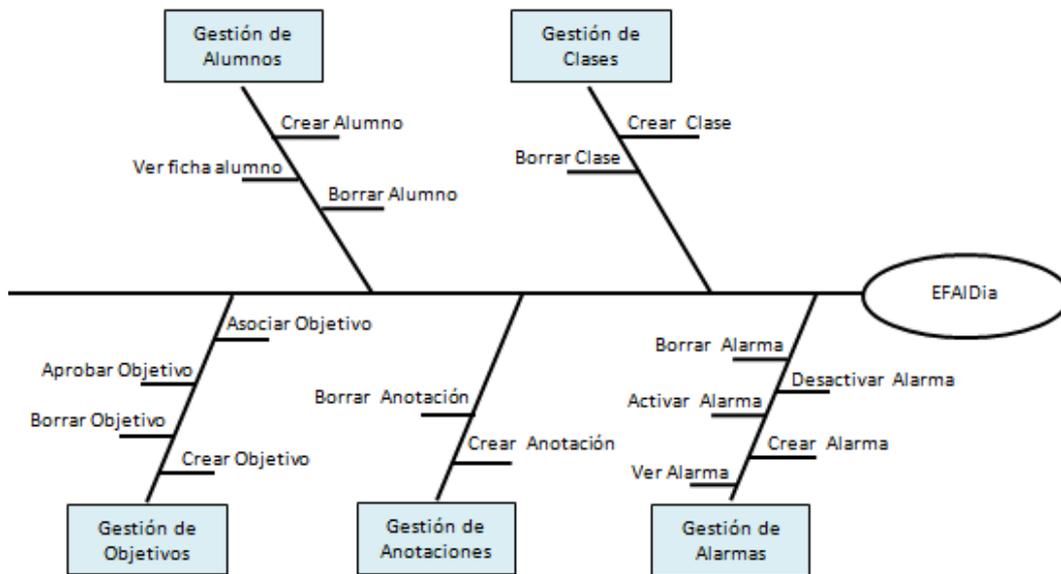


Figura 1.- Diagrama de características.

- Gestión de alumnos:

Este apartado detalla todas las acciones que el docente podrá realizar sobre los alumnos. Le permitirá crear, borrar, editar y visualizar la ficha del alumno, que contendrá toda su información personal además de los objetivos y anotaciones asociadas.

- Gestión de clases:

Este apartado detalla todas las acciones que el docente podrá realizar sobre las clases. Le permitirá crear, editar y borrar clases.

- Gestión de objetivos:

Este apartado detalla todas las acciones que el docente podrá realizar sobre los objetivos. Le permitirá crear, borrar, aprobar (si el alumno lo ha superado) y asociar objetivos (tanto a nivel global, incluyendo a todos los alumnos de una clase, como a nivel individual incluyendo únicamente a un alumno).

- Gestión de anotaciones:

Este apartado detalla todas las acciones que el docente podrá realizar sobre las anotaciones. Le permitirá crear, borrar y editar las anotaciones.

- Gestión de alarmas:

Este apartado detalla todas las acciones que el docente podrá realizar sobre las alarmas. Le permitirá crear, borrar, visualizar y activar/desactivar alarmas desde su listado o desde el calendario.

1.4. Reglas de negocio

Con relación al alcance y los objetivos definidos anteriormente, se establecen unas normas básicas que definen aspectos del diseño y funcionamiento de la aplicación móvil:

- **RN-01:** El tratamiento y almacenamiento de los datos sensibles se realizará de acuerdo a la Regulación General de Protección de Datos (GDPR).
- **RN-02:** El uso de la aplicación móvil será gratuita para todos los usuarios de la misma.

1.5. Organización del documento.

En este apartado realizaremos un análisis del contenido del documento, en el que describiremos los capítulos que lo forman para así facilitar su lectura.

La memoria de este trabajo está formada por tres secciones, que comprenden diez capítulos, cada uno formado por diferentes apartados que ayudan a mantener la correcta estructura del documento.

- **Sección 1 – Introducción.**
 - **Capítulo 1, Introducción:** este capítulo expondrá al lector el contexto del trabajo que se va a desarrollar. Está formado por seis apartados. En ellos veremos la motivación, objetivos y características que llevaron a desarrollar la APP. Continuará con las reglas de negocio y finalmente la organización del documento y del CD a entregar.
 - **Capítulo 2, Estado del arte:** en este capítulo se realizara un análisis y comparativa de los sistemas que existen actualmente en el mercado con funcionalidades similares a la aplicación presentada.
- **Sección 2 – Desarrollo.**
 - **Capítulo 3, Gestión del proyecto:** en este capítulo se realizan las estimaciones económicas y temporales necesarias para realizar el proyecto.
 - **Capítulo 4, Análisis del sistema:** en este capítulo se analiza el funcionamiento del sistema. El análisis se realiza desde tres puntos de vista, desde el del usuario (requisitos de usuario), desde el del sistema (requisitos funcionales y no funcionales) y desde el de los datos (requisitos de información).
 - **Capítulo 5, Diseño de Software:** en este capítulo encontraremos el comportamiento del sistema, veremos la arquitectura de la aplicación, tanto lógica como física.

- **Capítulo 6, Implementación:** en este capítulo veremos qué tecnologías y lenguajes han sido utilizados en la elaboración de la aplicación móvil.
- **Capítulo 7, Pruebas:** en este capítulo veremos las pruebas realizadas tanto durante como después de la creación de la aplicación móvil.
- **Sección 3 – Manuales.**
 - **Capítulo 8, Manuales:** este capítulo está formado por dos apartados. En el primero veremos el manual de instalación y posteriormente veremos el manual de uso de la aplicación.
- **Sección 4 – Conclusiones.**
 - **Capítulo 9, Conclusiones:** en este capítulo se expondrán las conclusiones obtenidas de la creación del sistema y los posibles cambios y mejoras que se podrían implementar en una futura versión.
 - **Capítulo 10, Bibliografía:** en este capítulo se mostrará los recursos utilizados durante el proyecto.

1.6. Contenido del CD-ROM

Junto a la memoria impresa, se entregará un disco que contendrá el código fuente de la aplicación y una copia de la memoria. La estructura del CD-ROM será la siguiente:

- **Memoria:** directorio que contendrá la memoria en formato PDF.
- **Código:** directorio que contendrá el código fuente proyecto.
- **Diagramas:** directorio que contendrá los diagramas usados en la memoria.

Capítulo 2 - Estado del arte

En la actualidad existen aplicaciones móviles similares, que permiten la gestión y organización de la actividad docente, alguna de estas aplicaciones son:

Aplicaciones que facilitan la comunicación y el aprendizaje:

- **CLASSDOJO:** aplicación utilizada por 2 de cada 3 colegios en Estados Unidos. Esta aplicación facilita la comunicación entre padres, profesores y alumnos, permitiendo a los alumnos publicar lo que han aprendido mediante videos y fotos. Además es usada como una herramienta motivacional con los estudiantes ya que los profesores les animaran con comentarios. Tiene muchas funcionalidades para ayudar a los docentes como:
 - *Group Maker:* genera grupos al azar entre los estudiantes de la clase, para la realización de diferentes actividades.
 - *Direcciones:* muestra instrucciones a seguir dentro del aula, así los estudiantes siempre sabrán en que deben trabajar.
 - *Música:* proporciona diferentes estilos de música que se adaptan a cada actividad, para enfocar a los alumnos al trabajo deseado.
 - *Medidor de ruido:* el medidor de ruido ayudará a los alumnos a conocer cuál es el volumen correcto del aula en momentos específicos.
 - *Pensar en pareja:* esta opción hará que los estudiantes tengan que trabajar con un compañero, fomentando así el trabajo en equipo.
 - *Aleatorio:* permite la selección de un estudiante al azar.
 - *Minutero:* reloj que permite a los estudiantes saber cuánto tiempo disponible les queda para completar la actividad en curso.
 - *Hoy:* permite la publicación del contenido que se impartirá en clase ese día, para que los alumnos estén informados de la actividad que se realizara.

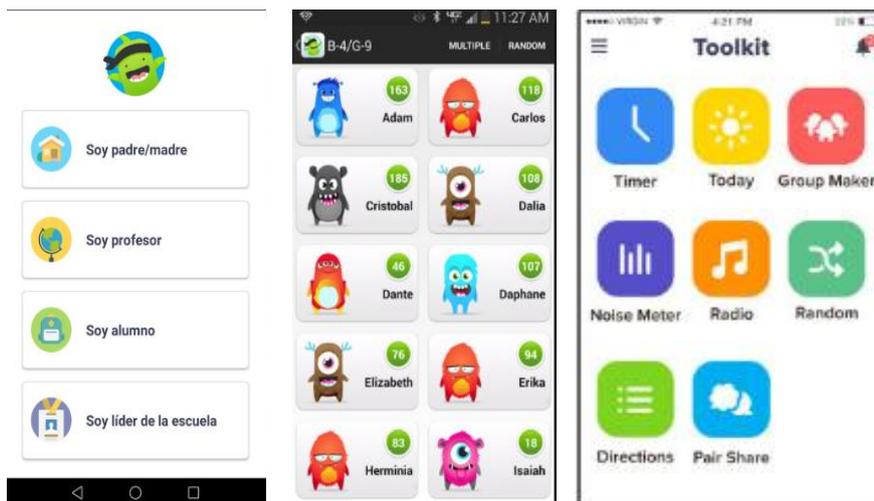


Figura 2.- Interfaz de “Class Dojo”.

En la Figura 2, se muestran 3 capturas de pantalla de la aplicación, donde vemos que la APP está diseñada para ser utilizada por distintos tipos de usuarios. También vemos un listado de alumnos de la clase en la que únicamente se muestran nombres sin fotografías personales y un menú con las múltiples acciones que permite esta aplicación, principalmente destinadas al entretenimiento de la clase, pero no a la recogida de información para la evaluación.

Esta herramienta es totalmente gratuita y está disponible tanto para ordenadores como para dispositivos iOS, Android y Kindle Fire.

- **ECOLAPP:** Esta aplicación es configurable para cada centro, por lo que cada uno de ellos podrá tener una APP propia y exclusiva. La herramienta tiene una parte genérica previamente configurada con múltiples funcionalidades.

Dispone de una plataforma de gestión online para el centro escolar, en la que se crean y actualizan los datos y una parte móvil que permite a los padres recibir avisos, autorizaciones de salida, notas consultas del menú diario, rutas de autobuses, etc. También les permite comprar libros, uniformes o contratar actividades extraescolares desde la aplicación.

Los profesores tienen un usuario especial para realizar sus tareas, mediante el cual, también pueden enviar archivos a sus alumnos. Además esta aplicación cuenta con una sección para el AMPA.

No es gratuita, su precio variará en función de la personalización que se requiera para adecuarla al centro en concreto. Por ello, tampoco disponemos de imágenes de su interfaz.

- **GoConqr:** aplicación para mejorar la experiencia de aprendizaje digital, pensada para estudiantes y profesores.

Entre los profesores permite la selección del grado en el que se imparte las clases (Primaria, Secundaria, Universidad, FP) y de la asignatura a impartir. Una vez indicados estos datos, la aplicación propone artículos y secciones adaptados a la selección realizada. También crea carpetas para publicar los contenidos de las clases que se impartan y ofrece la posibilidad de unirse a grupos similares a tus criterios.

Dispone de una biblioteca compartida, con recursos creados por los miembros de esta comunidad y también posee herramientas para, entre otros, la creación de diapositivas y test de evaluación. También permite crear alertas y círculos de amistades.

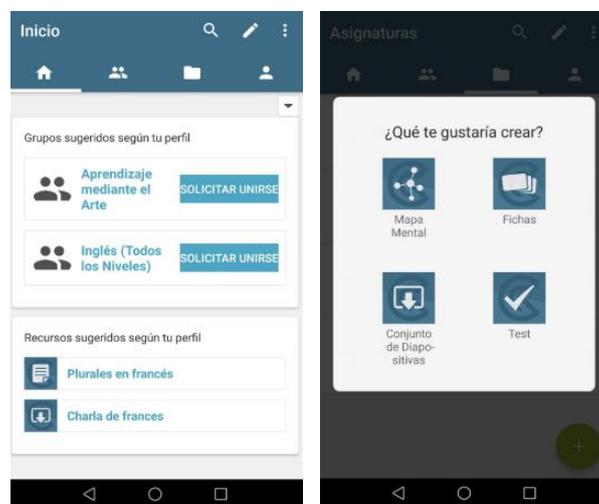


Figura 3.- Interfaz de “GoConqr”.

En la Figura 3 podemos ver dos capturas de pantalla de la aplicación, en la que se muestran las opciones de “unirse a grupos” para compartir contenidos de las clases y las funcionalidades tan diferentes a la aplicación que se va a desarrollar.

A diferencia de la anterior, esta aplicación es totalmente gratuita y está disponible para dispositivos iOS y Android. Para comenzar a utilizarla, requiere que el usuario realice un proceso de registro.

Aplicaciones que permiten la organización de actividades:

- **Teacher Aide Pro:** aplicación diseñada por profesores, actualmente disponible en su tercera versión e inicialmente gratuita. Para la primera clase que se dé de alta, se dispone de “funciones premium”, de tal manera que se pueden probar distintas funcionalidades/opciones añadidas y, si se desea, activar su suscripción mediante un pago mensual.

Las características principales de esta aplicación son:

- Control de asistencia.
- Libro de calificaciones.
- Plano del aula, permite ver dónde se sienta cada alumno.
- Identificar estudiantes en riesgo.
- Elaboración de informes PDF.
- Comunicación con los tutores mediante el envío de mensajes (sms o correos electrónicos) de forma masiva.
- Visualización de datos
- Tomar notas de estudiantes

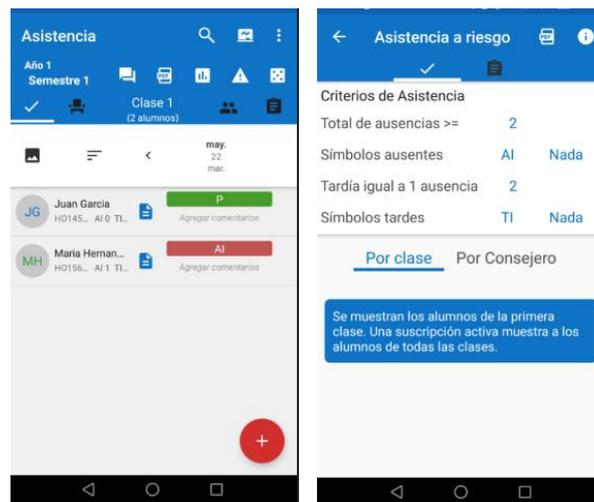


Figura 4.- Interfaz de “Teacher Aide Pro”.

En la Figura 4 vemos dos capturas de pantalla de la aplicación, en la que se muestra una lista de alumnos similar a la que se verá en la aplicación que se va a desarrollar. También se pueden ver las funcionalidades de asistencia y organización, que distan mucho de lo que se propone en este proyecto.

La aplicación está disponible para iOS y Android.

- **Cuaderno del profesor:** esta es una de las aplicaciones que más similitud tiene con la que se va a desarrollar. Creada por un profesor, permite llevar a cabo la gestión de una clase mediante múltiples funcionalidades:
 - Creación y gestión de notas de los alumnos. Pueden ser numéricas, tipo contador de texto, icono, valores y calculadas.
 - Evaluación mediante rúbricas y cálculo de nota promedio final.
 - Control de asistencia con plano del aula.
 - Exportación / Importación de datos.
 - Creación de horario.
 - Seguridad: copias sincronizadas con la nube y encriptación de la base de datos.
 - Diario de clase.
 - Creación y visionado de gráficas.
 - Realización de llamadas.

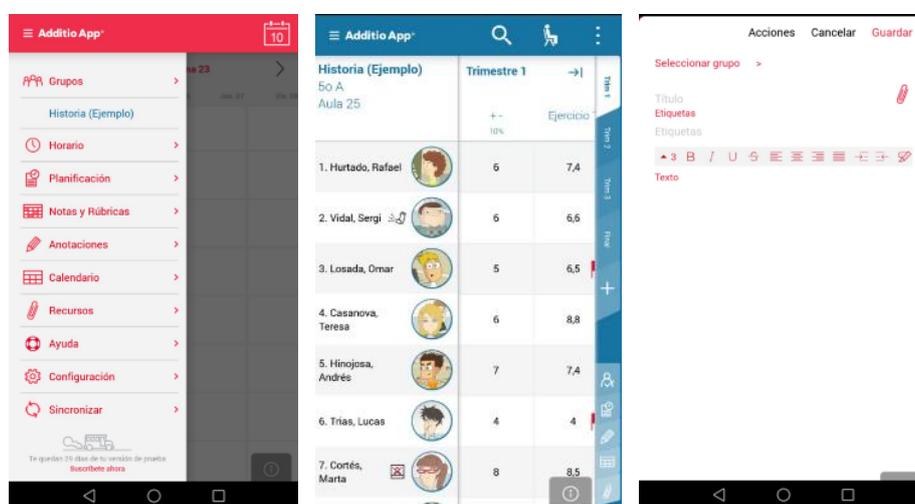


Figura 5.- Interfaz de “Cuaderno del profesor”.

En la Figura 5 podemos ver las funcionalidades de las que dispone la aplicación, una lista de alumnos y la pantalla de toma de notas, similar a la aplicación a desarrollar. Al realizar la instalación y durante el primer uso, dispone de un tutorial para ayudar al docente a comprender las funcionalidades de las que puede hacer uso.

La aplicación es gratuita, pero contiene publicidad. Ésta puede eliminarse haciéndose usuario premium.

2.1. Conclusiones del estado del arte

El valor añadido que proporciona EFAIDia es que está diseñada y enfocada principalmente para la actividad docente de educación física. El primer bloque de aplicaciones descritas tiene una labor más administrativa y comunicativa padre-profesor, alumno-profesor. El segundo bloque tiene una labor más organizativa.

La aplicación propuesta realiza una labor más docente, unificará partes de ambos tipos de aplicaciones, permitiendo al docente gestionar de manera específica las clases de educación física, programar de manera más rápida y eficaz sus sesiones y facilitar la evaluación en cualquier momento.

A continuación se expone una tabla que comparará las características de las aplicaciones anteriormente descritas con respecto a las de la aplicación a desarrollar, para poder detectar similitudes y diferencias y hacer énfasis en lo que caracteriza EFAIDia.

<i>Funcionalidad / Aplicación</i>	<i>CLASS DOJO</i>	<i>ECOLAP</i>	<i>GoConqr</i>	<i>Teacher Aide</i>	<i>Cuaderno del Profesor</i>	<i>EFAIDia</i>
<i>Gratuita</i>	<i>Si</i>	<i>No</i>	<i>Si</i>	<i>No</i>	<i>Si</i>	<i>Si</i>
<i>Comunicación (padre/profesor)</i>	<i>Si</i>	<i>Si</i>	<i>No</i>	<i>Si</i>	<i>No</i>	<i>No</i>
<i>Publicar contenido a impartir (con Alumnos)</i>	<i>Si</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>
<i>Interacción con alumnos (Creación de grupos, música...)</i>	<i>Si</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>
<i>Multiusuario</i>	<i>Si</i>	<i>Si</i>	<i>Si</i>	<i>Si</i>	<i>No</i>	<i>No</i>
<i>Comunicación con la comunidad docente.</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>No</i>	<i>No</i>	<i>No</i>
<i>Permite calificaciones.</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>Si</i>	<i>Si</i>	<i>Si</i>
<i>Creación de documentos para clases.</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>Si</i>	<i>No</i>	<i>No</i>
<i>Control de asistencia</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>No</i>	<i>No</i>
<i>Tomar notas de alumnos</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>Si</i>	<i>Si</i>
<i>Exportación/Importación datos</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>Si</i>
<i>Creación de horario</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>Si</i>
<i>Llamadas</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>No</i>
<i>Creación de graficas</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>	<i>No</i>
<i>Planificación de sesiones</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>
<i>Gestión de alarmas/eventos</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>
<i>Personalización ficha alumno</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>No</i>	<i>Si</i>

Tabla 1.- Comparación de características de las aplicaciones.

Como se puede observar en la tabla, la herramienta a desarrollar no está centrada en la comunicación ni en la enseñanza hacia el alumno, si no que está destinada a un único usuario, el docente, con el fin de facilitar su trabajo de evaluación.

Esta herramienta es la única que permite desglosar el contenido de la clase en pequeños objetivos evaluables y que permite la gestión de alarmas. De todas las herramientas expuestas es la única diseñada para docentes de la materia de educación física, por lo que la plataforma elegida es un dispositivo móvil, con el fin de que el docente tenga facilidad de movimiento y pueda seguir todas las actividades de los alumnos, perdiendo el mínimo tiempo en la toma de datos.

SECCIÓN 2 – DESARROLLO

Capítulo 3 - Gestión del proyecto

En este apartado se realizará la planificación del proyecto, antes de comenzar a realizar el análisis del mismo. Se describe la metodología a seguir, la planificación temporal en la que se dividirá el proyecto, una estimación presupuestaria y por último, el coste real del desarrollo.

3.1. Metodología

Para el desarrollo del software, se ha seguido una metodología en cascada, por la cual se distribuirá el trabajo en diferentes etapas. Cada etapa comenzará con la finalización de la anterior, salvo la etapa de documentación, que se desarrollara a lo largo de todo el proceso. La etapa de pruebas formará parte de la implementación, debido a que las pruebas serán realizadas a medida que se vaya desarrollando el producto.

Cualquier error encontrado en la fase de pruebas obligará necesariamente a continuar con el desarrollo del producto para su corrección. Esta metodología proporciona una forma de trabajo más efectiva, ya que en este proyecto, el diseño y los requisitos están definidos previamente por los docentes de educación física. Por ello, ambas etapas de diseño y definición de requisitos están ya establecidas y no serán modificables.

Las fases utilizadas serán las siguientes:

- **Fase 1:** Definición y requisitos.
- **Fase 2:** Diseño de herramienta.
- **Fase 3:** Diseño de la interfaz de usuario.
- **Fase 4:** Desarrollo.
- **Fase 5:** Pruebas de funcionamiento de la aplicación.
- **Fase 6:** Documentación.

3.2. Fases de trabajo y estimación temporal

Como se ha indicado anteriormente, el desarrollo de esta herramienta se realizará siguiendo un modelo en cascada. El tiempo se distribuirá en las siguientes fases:

- **Fase 1, Definición y requisitos:** En esta fase se analizan y definen las necesidades del usuario, obteniendo todos los requisitos que la aplicación deberá cumplir. Para la elaboración de esta etapa se ha contado con una alumna del Grado de Magisterio en Educación Física.
- **Fase 2, Diseño de herramienta:** Se definirá la estructura interna del software. Esta etapa está centrada en establecer la distribución que deberá tener la aplicación y valorar si será necesario el uso de recursos externos en función de las necesidades descritas anteriormente por el usuario.
- **Fase 3, Diseño de la interfaz de usuario:** durante esta fase se diseña la interfaz, teniendo en cuenta el tipo de usuario al que está destinado, para adecuar su imagen, que sea usable y funcional. Para la elaboración de esta etapa se ha contado de nuevo con una alumna del Grado de Magisterio en Educación Física. Además se han realizado diferentes pruebas con prototipos, con la colaboración de profesores del colegio Villalpando.
- **Fase 4, Desarrollo:** se implementará el código fuente necesario para cumplir con los requisitos especificados en la fase 1, utilizando las herramientas indicadas en el diseño. En esta fase se realizará la programación de algoritmos que permitan que la aplicación tenga la funcionalidad definida anteriormente.
- **Fase 5, Pruebas de funcionamiento de la aplicación:** esta es la fase de testeo de la aplicación, se probarán todas las funcionalidades implementadas. En este caso las pruebas se realizarán a medida que se desarrolle el producto y, una vez la funcionalidad este testeada y finalizada, se pasará a la siguiente. De lo contrario, se continuará con la corrección de la funcionalidad en desarrollo.
- **Fase 6, Documentación:** esta etapa consiste en redactar todo lo que se ha llevado a cabo en las anteriores fases. Se elaborará una documentación cuyo objetivo será tener informadas las funcionalidades que realiza el sistema, para facilitar su uso y posteriores modificaciones. Esta documentación se llevará a cabo durante toda la elaboración de la aplicación.

En la siguiente tabla reflejamos cada una de las fases, mostrando la duración temporal estimada:

		Nombre	Duración	Inicio	Terminado
1		Definición y requisitos	7 days?	18/12/17 19:00	29/12/17 18:00
2		Diseño de la herramienta	14 days?	4/01/18 19:00	24/01/18 20:00
3		Diseño de la interfaz de usuario	21 days?	24/01/18 20:00	2/03/18 17:00
4		Desarrollo	56 days?	2/03/18 17:00	6/06/18 21:00
5		Pruebas de funcionamiento	14 days?	6/06/18 21:00	24/06/18 15:00
6		Documentación	110 days?	27/12/17 19:00	26/06/18 20:00

Tabla 2.- Estimación temporal de las fases del desarrollo.

Para ilustrar esta tabla de forma grafica se usara un diagrama de Gantt, que nos permite observar más rápidamente como se desarrollaran las fases.

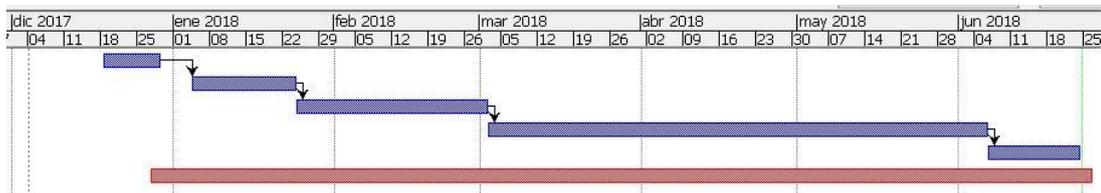


Figura 6.- Diagrama de Gantt.

Como se puede observar en el diagrama, se desarrollará una fase tras otra, ya que el trabajo será desempeñado por una única persona. La fase de documentación acompaña todo el proceso, considerándola como parte del resto de fases.

3.3. Estimaciones

Para realizar el cálculo del presupuesto necesario para el desarrollo del proyecto, primero se deben realizar las estimaciones necesarias, por lo que se utilizarán dos modelos:

- El modelo por Puntos de Función (PF).
- El modelo COCOMO.

Al finalizar la estimación mediante dichos modelos, se realizará el presupuesto del trabajo.

3.3.1. Estimación de costes por Puntos de Función (PF)

Este método consiste en medir el tamaño del sistema dividiéndolo en unidades o componentes, según su grado de complejidad. Realizando el sumatorio resultante de sus valoraciones obtendremos un número de puntos de función que se multiplicará por un factor cuyo valor dependerá de las características del sistema.

Esto nos llevará a obtener un valor de puntos de función. Con esta puntuación se calculará el coste que tiene la realización del proyecto.

Pasos a seguir para realizar su cálculo:

- Se definen los dominios de información y su complejidad:
 - **Número de entradas:** datos que el usuario aporta al sistema (procesos de actualización e inserción de datos).
 - **Número de salidas:** datos que el sistema aporta al usuario (procesos de lectura y envío de datos al exterior del sistema).
 - **Número de consultas externas:** entradas que requieren una respuesta del sistema (procesos que combinan entradas y salidas).
 - **Numero de ficheros lógicos internos:** ficheros que únicamente utiliza el sistema (ficheros, bases de datos, etc.).
 - **Numero de ficheros lógicos externos:** ficheros usados por otros sistemas (ficheros, bases de datos de servidor, etc.).
- Se contarán el número de elementos de cada clase distinguiendo según su complejidad (alta, media o baja).
- Obtención de los puntos medios de función no ajustados (PFNA), estos se obtendrán sumando las cantidades definidas anteriormente con los pesos de la siguiente tabla.

Dominio de información	Complejidad baja	Complejidad media	Complejidad alta
Entradas	x3	x4	x6
Salidas	x4	x5	x7
Consultas Externas	x3	x4	x6
Fich. Lógicas internas	x7	x10	x15
Fich. Lógicas externas	x5	x7	x10

Tabla 3.- Pesos de los dominios de información según complejidad.

- Tras definir los PFNA se deben ajustar a través de un factor de ajuste (FA) para con ello obtener los puntos de función ajustados (PFA)
- Por otra parte, para calcular las líneas de código se utiliza una tabla de equivalencias. A continuación se detalla con los lenguajes utilizados en el desarrollo.

Lenguaje	LDC/PF
Java (Android)	53

Tabla 4.- Estimación de las líneas de código por lenguaje de programación.

3.3.1.1. Estimación del desarrollo.

- Número de entradas:

Nombre	Complejidad
Formulario de registro	Baja
Formulario de login	Baja
Formulario creación alarma	Baja
Formulario creación clase	Media
Formulario edición clase	Media
Formulario creación alumno	Media
Formulario edición alumno	Media
Formulario creación objetivo	Baja
Formulario creación anotación	Media
Formulario edición anotación	Baja

Tabla 5.- Número de entradas.

- Número de salidas:

Nombre	Complejidad
Pantalla ficha alumno	Baja
Pantalla clase	Baja
Pantalla listado clases	Baja
Pantalla listado alarmas	Baja
Pantalla horario	Media
Pantalla calendario	Media
Exportación datos	Media

Tabla 6.- Número de salidas.

- Número de consultas externas:

Nombre	Complejidad
Consulta de alumnos	Baja
Consulta de objetivos	Baja
Consulta de anotaciones	Baja
Consulta de clases	Baja
Consulta de horarios	Baja

Tabla 7.- Número de consultas externas.

- Numero de ficheros lógicos internos:

Nombre	Complejidad
Base de datos de la aplicación	Baja

Tabla 8.- Número de ficheros lógicos internos.

- Numero de ficheros lógicos externos

Nombre	Complejidad
Google "SpeechToText"	Baja
Ficheros importaciones	Media

Tabla 9.- Número de ficheros lógicos externos.

- Cálculo:

Dominio de Información	Complejidad	Total por complejidad	Suma de complejidades
Entradas	Baja	5x3	35
	Media	5x4	
	Alta	0x6	
Salidas	Baja	4x4	31
	Media	3x5	
	Alta	0x7	
Consultas externas	Baja	5x3	15
	Media	0x4	
	Alta	0x6	
Ficheros lógicos internos	Baja	1x7	7
	Media	0x10	
	Alta	0x15	
Ficheros lógicos externos	Baja	1x5	12
	Media	1x7	
	Alta	0x10	
Total de puntos de función (PFNA)			100

Tabla 10.- Cálculo de puntos de función no ajustados.

- Cálculo del factor de ajuste:

Factor de complejidad	Complejidad/peso
Comunicación de datos	3
Funciones distribuidas	0
Rendimiento	3
Gran carga de trabajo	3
Frecuencia de transiciones	3
Entrada on-line de datos	3
Requisitos de manejo del usuario final	4
Actualizaciones online	2
Procesos complejos	2
Utilizacion de otros sistemas	2
Facilidad de mantenimiento	1
Facilidad de operacion	1
Instalacion en multiples lugares	0
Facilidad de cambio	1

Tabla 11.- Cálculo del factor de ajuste.

$$FA = 0,65 + (0,01 * 28) = \mathbf{0,93}$$

- Calculo de los puntos de función ajustados:

$$PFA = 100 * 0,93 = \mathbf{93}$$

Para realizar el cálculo de las líneas de código (en adelante LDC o KLDC si se contabilizan 1000 líneas) de la aplicación móvil, se utiliza la equivalencia del lenguaje Java (Android) especificada en la tabla 4, 1PF = 53 líneas de código.

$$LDC = 53 * 93 = \mathbf{4.929} \text{ LDC}$$

Este valor será utilizado posteriormente para realizar el cálculo de la estimación nominal en el método COCOMO.

3.3.2. Estimación de costes por COCOMO

Esta estimación es utilizada para medir el esfuerzo y el tiempo que costará realizar la aplicación. Este método se centra en la estimación previa del tamaño del sistema en líneas de código (LDC).

COCOMO tiene 3 métodos, nosotros usaremos el COCOMO intermedio ya que nuestro proyecto se encuentra en las últimas fases del desarrollo.

A su vez este método comprende tres modelos (orgánico, semi-acoplado e integrado), dependiendo de tamaño y complejidad. En este caso usaremos el semi-acoplado ya que es una aplicación sencilla pero con los requisitos bien definidos.

Según el tipo de proyecto deberemos tener en cuenta las siguientes constantes:

Proyecto software	A	B	C	D
Orgánico	2,4	1,05	2,5	0,38
Semi-acoplado	3,0	1,12	2,5	0,35
Integrado	3,6	1,20	2,5	0,32

Tabla 12.- Constantes para cada modelo COCOMO.

Además, para ajustar los factores de coste, existe la siguiente tabla:

Atributos	Valor					
	Muy bajo	Bajo	Nominal	Alto	Muy alto	Extra alto
Atributos de software						
Fiabilidad	0,75	0,88	1,00	1,15	1,40	
Tamaño de Base de datos		0,94	1,00	1,08	1,16	
Complejidad	0,70	0,85	1,00	1,15	1,30	1,65
Atributos de hardware						
Restricciones de tiempo de ejecución			1,00	1,11	1,30	1,66
Restricciones de memoria virtual			1,00	1,06	1,21	1,56
Volatilidad de la máquina virtual		0,87	1,00	1,15	1,30	
Tiempo de respuesta		0,87	1,00	1,07	1,15	
Atributos de personal						
Capacidad de análisis	1,46	1,19	1,00	0,86	0,71	
Experiencia en la aplicación	1,29	1,13	1,00	0,91	0,82	
Calidad de los programadores	1,42	1,17	1,00	0,86	0,70	
Experiencia en la máquina virtual	1,21	1,10	1,00	0,90		
Experiencia en el lenguaje	1,14	1,07	1,00	0,95		
Atributos del proyecto						
Técnicas actualizadas de programación	1,24	1,10	1,00	0,91	0,82	
Utilización de herramientas de software	1,24	1,10	1,00	0,91	0,83	
Restricciones de tiempo de desarrollo	1,22	1,08	1,00	1,04	1,10	

Tabla 13.- Factores de coste del esfuerzo.

Las formulas utilizadas son las siguientes:

- Esfuerzo nominal (personas/mes)

$$A * (KLDC)^B$$

- Esfuerzo (personas/mes)

$$Esfuerzo Nominal * \pi Factores de coste$$

- Tiempo de desarrollo (meses)

$$2,5 * Esfuerzo^C$$

- Número medio de personas (personas)

$$\frac{Esfuerzo}{Tiempo de desarrollo}$$

Obtendremos los KLDC dividiendo las LDC/1000. Los datos A, B y C se obtendrán de la tabla 12.

3.3.2.1. Estimación del desarrollo

El resultado obtenido de la estimación por Puntos de Función fue 4.929 LDC, lo que equivale a 4,92 KLDC. Con este valor comenzamos la estimación mediante COCOMO.

$$\text{Esfuerzo nominal} = 3,0 * (4,92)^{1,12} = 17,86 \approx \mathbf{18 \text{ personas/mes}}$$

$$\text{Esfuerzo} = 17,86 * 0,91 \text{ (Técnicas actualizadas de programación)} * 0,95 \text{ (Experiencia en el lenguaje)} * 0,91 \text{ (Experiencia en la aplicación)} = \mathbf{14,05 \text{ personas/mes}}$$

$$\text{Tiempo de desarrollo} = 2,5 * 14,05^{0,35} = 6,3 \approx \mathbf{6 \text{ meses}}$$

$$\text{Numero medio de personas} = 17,86/6,3 = 2,83 \approx \mathbf{3 \text{ personas}}$$

Se necesitarían 3 personas durante 6 meses para realizar la aplicación móvil. Como lo realizara una única persona saldría un total 18 meses

3.4. Presupuestos

3.4.1. Presupuesto inicial

Se estima para el presupuesto inicial una duración de 6 meses para una única persona. Esta estimación difiere de lo calculado en COCOMO ya que esa estimación no se ajusta a la realidad.

3.4.1.1. Presupuesto hardware

Para el desarrollo que se va a realizar se estiman necesarios los siguientes componentes hardware.

- **Ordenador personal:** para realizar las fases de análisis, diseño, implementación y pruebas.
- **Conexión a Internet:** para la búsqueda e investigación.
- **Impresora:** para realizar la impresión del documento definitivo.

Componente	Uso	Coste Total(€)	Coste final(€)
Ordenador	70%	600	420
Conexión	6 meses	28	168
Impresora	10%	50	5
Material de oficina	100%	10	10
			Total : 603€

Tabla 14.- Presupuesto Hardware.

3.4.1.2. Presupuesto software

Componente	Uso(%)	Coste Total(€)	Coste final(€)
Windows	70	20	14
Android Studio	65	0	0
OpenProj	10	0	0
Microsoft Office	1	13	13
Star UML	5	0	0
			Total : 27€

Tabla 15.- Presupuesto Software.

3.4.1.3. Presupuesto de desarrollo

La planificación temporal inicial es de 6 meses, trabajo que realizará una persona. Durante estos 6 meses se trabajarán 5 horas diarias los 22 días laborales que forman un mes, por lo que el cálculo de la mano de obra es el siguiente:

Horas totales de trabajo = 6 meses * 5 horas * 22 días al mes = 660 horas.

Trabajador	Horas	Coste/hora(€)	Coste final(€)
Ingeniero técnico	660	8	5.280
			Total : 5.280€

Tabla 16.- Presupuesto de desarrollo.

3.4.1.4. Presupuesto total

Para el cálculo del presupuesto total se sumaran el presupuesto de los componentes Hardware y Software además de la mano de obra.

Componente	Coste(€)
Hardware	603
Software	27
Trabajador	5.280
Total : 5.910€	

Tabla 17.- Presupuesto total.

3.4.2. Presupuesto final

En la realización de este presupuesto se ha tenido en cuenta la duración real del trabajo llevado a cabo.

3.4.2.1. Presupuesto hardware

Para el desarrollo final se han utilizado los componentes hardware que se plantearon al inicio:

- **Ordenador personal:** para realizar las fases de análisis, diseño, implementación y pruebas.
- **Conexión a Internet:** para la búsqueda e investigación.
- **Impresora:** para realizar la impresión del documento definitivo.

Componente	Uso	Coste Total(€)	Coste final(€)
Ordenador	70%	600	420
Conexión	6 meses	28	168
Impresora	10%	50	5
Material de oficina	100%	10	10
			Total : 603€

Tabla 18.- Presupuesto Hardware.

3.4.2.2. Presupuesto software

Para la elaboración final de la aplicación móvil se han necesitado los elementos hardware indicados en el presupuesto inicial:

Componente	Uso(%)	Coste Total(€)	Coste final(€)
Windows	70	20	14
Android Studio	65	0	0
OpenProj	10	0	0
Microsoft Office	1	13	13
Star UML	5	0	0
			Total : 27€

Tabla 19.- Presupuesto Software.

3.4.2.3. Presupuesto de desarrollo

La duración final del trabajo ha sido de 7 meses, un mes más de lo planificado inicialmente. Añadiendo durante los 22 días laborables del mes añadido un total de 5 horas al día, se obtiene el siguiente cálculo:

Horas totales de trabajo = (6 meses * 5 horas * 22 días al mes) + (1 mes * 5 horas * 22 días) = 770 horas

Trabajador	Horas	Coste/hora(€)	Coste final(€)
Ingeniero técnico	770	8	6.160
			Total : 6.160€

Tabla 20.- Presupuesto de desarrollo.

3.4.2.4. Presupuesto final

Para calcular el presupuesto final se sumaran los componentes software y hardware utilizados en el desarrollo además de la mano de obra final.

Componente	Coste
Hardware	603
Software	27
Trabajador	6.160
	Total: 6.790 €

Tabla 21.- Presupuesto Final.

3.5. Conclusiones

Si se hubieran usado los datos obtenidos en las estimaciones realizadas a través del cálculo de los puntos de función y COCOMO, el presupuesto habría sido mucho mayor. Debido a que el lenguaje utilizado proporciona gran cantidad de librerías, y ya que en esta aplicación se podrá reutilizar multitud de líneas de código fuente, se puede observar una variación entre la estimación realizada por los métodos PFA/COCOMO y la estimación inicial/final.

Finalmente ha habido una desviación de un mes (entre el presupuesto inicial y final) por que han aumentado los tiempos del desarrollo. Esta desviación únicamente ha afectado al presupuesto de desarrollo, ya que los componentes hardware y software utilizados han sido los mismos.

Capítulo 4 - Análisis del sistema

En este apartado se describe el análisis del sistema. Se especificarán los tipos de usuario que tendrá la aplicación, los requisitos que cumplirá la herramienta y los diagramas de casos de uso.

4.1. Actores del sistema

Esta aplicación es para profesores, pero no está pensada para manejarla a nivel global de un colegio, sino particularmente por un profesor. Por lo que únicamente se contempla un actor que interactúa con el sistema.

ACT-1	Usuario genérico
Versión	1.0(02/04/2018)
Autor	Clara María Herranz de Andrés
Descripción	Toda persona que utiliza la aplicación de manera estándar para guardar, visualizar información y organizar una clase de educación física
Comentarios	El usuario tendrá un registro únicamente para la seguridad de la información manejada.

Tabla 22.- Usuario genérico.

4.2. Requisitos de usuario

A continuación se especificarán los requisitos de usuario, que definen las características que deberá poseer el sistema para cumplir las necesidades del mismo.

RU-01: El usuario podrá registrarse en la aplicación aportando correo electrónico usuario y contraseña.

RU-02: El usuario podrá iniciar sesión en la aplicación aportando su usuario y contraseña.

RU-03: El usuario identificado podrá cerrar sesión

RU-04: El usuario podrá solicitar la recuperación de la contraseña.

RU-05: El usuario identificado podrá visualizar el calendario

RU-06: El usuario identificado podrá consultar su horario de clases.

RU-07: El usuario identificado podrá crear su horario de clases.

RU-08: El usuario identificado podrá crear clases.

RU-09: El usuario identificado podrá consultar su listado de clases.

RU-10: El usuario identificado podrá editar una clase.

RU-11: El usuario identificado podrá crear alumnos.

RU-12: El usuario identificado podrá eliminar alumnos.

RU-13: El usuario identificado podrá consultar los alumnos de sus clases.

RU-14: El usuario identificado podrá importar un conjunto de alumnos.

RU-15: El usuario identificado podrá editar un alumno.

RU-16: El usuario identificado podrá consultar la ficha de un alumno.

RU-17: El usuario identificado podrá crear alarmas.

RU-18: El usuario identificado podrá consultar sus alarmas.

RU-19: El usuario identificado podrá activar y desactivar alarmas.

RU-20: El usuario identificado podrá eliminar alarmas.

RU-21: El usuario identificado podrá crear objetivos.

RU-22: El usuario identificado podrá asignar objetivos a un alumno.

RU-23: El usuario identificado podrá asignar objetivos a una clase, almacenándose en todos sus alumnos.

RU-24: El usuario identificado podrá eliminar objetivos de un alumno.

RU-25: El usuario identificado podrá editar objetivos.

RU-26: El usuario identificado podrá aprobar objetivos del alumno.

RU-27: El usuario identificado podrá crear anotaciones de texto en el alumno.

RU-28: El usuario identificado podrá crear anotaciones de voz que serán transcritas en el alumno.

RU-29: El usuario identificado podrá eliminar anotaciones del alumno.

RU-30: El usuario identificado podrá editar anotaciones del alumno.

RU-31: El usuario identificado podrá exportar datos de los alumnos.

RU-32: El usuario identificado podrá borrar los datos de la aplicación.

RU-33: El usuario identificado podrá importar un conjunto de clases.

RU-34: El usuario identificado podrá importar un conjunto de objetivos.

4.2.1. Diagrama de casos de uso

Una vez descritas las necesidades de los docentes de educación física en el apartado anterior, se listan los casos de uso que provienen de ellos. El usuario descrito en la figura 7 es el docente indicado en el Apartado 4.1.

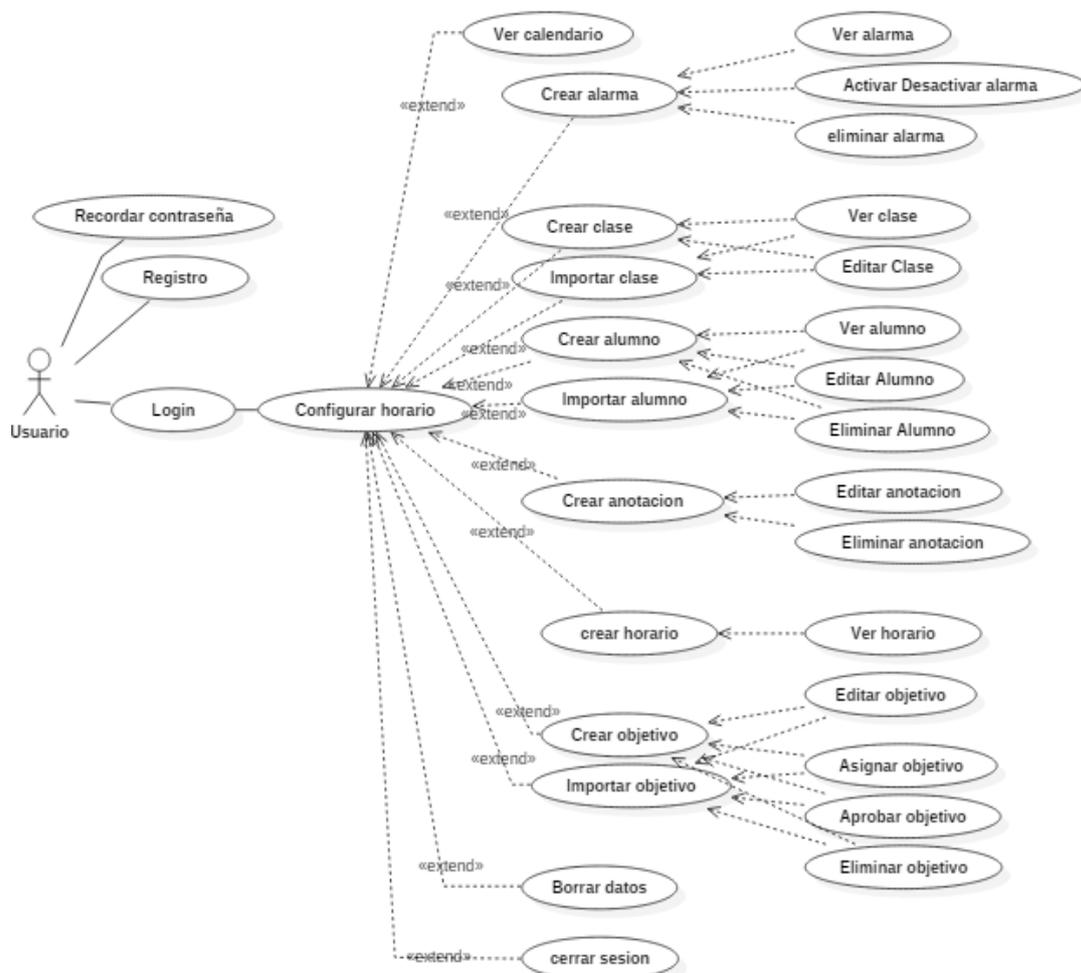


Figura 7.- Diagrama de Casos de Uso.

4.2.2. Especificación de casos de uso

- CU-01:** Registro.
CU-02: Login.
CU-03: Cierre de Sesión.
CU-04: Recuperación contraseña.
CU-05: Configuración horario.
CU-06: Visualizar calendario.
CU-07: Crear alarma.
CU-08: Visualizar alarma.
CU-09: Activar/Desactivar alarma.
CU-10: Eliminar alarma.
CU-11: Crear clase.
CU-12: Importar clase.
CU-13: Visualizar clase.
CU-14: Editar clase.
CU-15: Crear alumno.
CU-16: Importar alumno.
CU-17: Visualizar ficha alumno.
CU-18: Editar alumno.
CU-19: Eliminar alumno.
CU-20: Crear anotación.
CU-21: Editar anotación.
CU-22: Eliminar anotación.
CU-23: Visualizar horario.
CU-24: Crear objetivo.
CU-25: Importar objetivo.
CU-26: Asignar objetivo.
CU-27: Aprobar objetivo.
CU-28: Eliminar objetivo.
CU-29: Borrar datos.

CU-01		Registro	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario registrarse en la aplicación.		
Precondiciones	Solo se permitirá el registro la primera vez que se abra la aplicación.		
Secuencia Normal	Paso	Acción	
	2	El sistema muestra la pantalla con los campos a completar por el usuario.	
	3	El usuario rellena los datos y acepta el registro.	
Secuencia alternativa	4	El sistema recoge y almacena los datos.	
	Paso	Acción	
Postcondición	Los datos del usuario quedan guardados en la base de datos.		
Excepciones	Paso	Acción	
	3	El usuario no completa los datos correctamente.	
	4	El sistema informa de los datos que faltan por completar y permanecen en la pantalla de registro hasta que cancele o complete los datos pendientes.	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios	Un usuario registrado no podrá acceder nuevamente a la pantalla de registro por lo que no se podrá repetir el paso de registro.		

Tabla 23.- CU-01: Registro.

CU-02 Login		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario iniciar sesión en la aplicación.	
Precondiciones		
Secuencia Normal	Paso	Acción
	1	El usuario introduce sus datos en la pantalla de login del sistema.
	2	El sistema comprueba los campos.
	3	El sistema inicia sesión la sesión de usuario en la aplicación.
Secuencia alternativa	Paso	Acción
Postcondición	El usuario inicia sesión.	
Excepciones	Paso	Acción
	2	El usuario no introduce los datos correctamente.
	3	El sistema informa que los datos introducidos no son correctos y mantiene al usuario en la pantalla de login.
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 24.- CU-02: Login.

CU-03 Cierre de Sesión		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario cerrar sesión en la aplicación.	
Precondiciones		
Secuencia Normal	Paso	Acción
	1	El usuario pulsa sobre la opción del menú “cerrar sesión”.
	2	El sistema cierra la sesión del usuario y lo lleva a la pantalla del login.
Secuencia alternativa	Paso	Acción
Postcondición	Se cierra la sesión en el sistema	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 25.- CU-03: Cierre de Sesión.

CU-04 Recuperación contraseña		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario recordar su contraseña en caso de olvido.	
Precondiciones	Registro previo realizado	
Secuencia Normal	Paso	Acción
	1	El usuario pulsa sobre el botón enlace “recuperar contraseña”
	2	El sistema muestra un pop up informando que se envió un correo recordando la contraseña.
Secuencia alternativa	Paso	Acción
Postcondición	Email enviado al correo del usuario.	
Excepciones	Paso	Acción
	2	El sistema muestra un mensaje “Error al enviar mail” debido a un error de conexión, credenciales etc.
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios	Hasta que el usuario no se registra en la aplicación este enlace no aparecerá en la pantalla de Login.	

Tabla 26.- CU-04: Recuperación contraseña.

CU-05 Configuración horario.		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario crear el horario sobre el que se representaran las clases.	
Precondiciones	Será en el primer inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	Al iniciar sesión por primera vez se abrirá la pantalla para configurar el horario.
	2	El usuario introducirá el horario del colegio (sus franjas horarias y recreos)
	3	El sistema creará su horario de clases y lo llevará a la pantalla principal.
Secuencia alternativa	Paso	Acción
Postcondición	El horario quedará almacenado en el sistema	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios	Este es el primer paso que deberá dar el usuario al entrar en la aplicación. Solo se permite la configuración del horario tras el primer registro.	

Tabla 27.- CU-05: Configuración horario.

CU-06	Visualizar calendario	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario ver su calendario que contendrá las alarmas planificadas.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario pulsa sobre el botón "calendario".
	2	El sistema muestra al usuario la pantalla del calendario.
Secuencia alternativa	Paso	Acción
Postcondición	El sistema mostrara la pantalla del calendario.	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 28.- CU-06: Visualizar calendario.

CU-07	Crear alarma	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario crear alarmas.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario pulsa sobre el botón de calendario.
	2	El sistema muestra la pantalla del calendario
	3	El usuario selecciona en el calendario el día sobre el que quiere crear la alarma.
	4	El sistema muestra la pantalla de creación de alarma.
	5	El usuario completa los campos solicitados y los guarda.
	6	El sistema registra la alarma y regresa a la pantalla del calendario, sobre el día pone un punto indicando que tiene una alarma.
Secuencia alternativa	Paso	Acción
	5	El usuario no completa los campos correctamente
	6	El sistema envía mensaje informando del error.
Postcondición	La alarma quedará almacenada en el sistema.	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Media	
Prioridad	Media	
Comentarios		

Tabla 29.- CU-07: Crear alarma.

CU-08		Visualizar alarma
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario ver las alarmas creadas.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	Desde cualquier pantalla, el usuario deberá seleccionar el icono “despertador” del menú superior.
	2	El sistema mostrará la pantalla con el listado de alarmas
Secuencia alternativa	Paso	Acción
Postcondición	Las alarmas serán mostradas en el sistema.	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Media	
Prioridad	Media	
Comentarios		

Tabla 30.- CU-08: Visualizar alarma.

CU-09		Activar/Desactivar alarma
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario activar y desactivar sus alarmas.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	Desde cualquier pantalla, el usuario deberá seleccionar el icono “despertador” del menú superior.
	2	El sistema mostrará la pantalla con el listado de alarmas
	3	El usuario desde un “botón deslizable” activará o desactivará las alarmas deseadas
	4	El sistema guardará el estado de la alarma, para su posterior interpretación desde el calendario.
Secuencia alternativa	Paso	Acción
Postcondición	El estado de la alarma quedará actualizado en el sistema.	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Media	
Prioridad	Media	
Comentarios	Este estado es usado al dibujar el calendario.	

Tabla 31.- CU-09: Activar/Desactivar alarma.

CU-10	Eliminar alarma	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario eliminar sus alarmas.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	Desde cualquier pantalla, el usuario deberá seleccionar el icono “despertador” del menú superior.
	2	El sistema mostrará la pantalla con el listado de alarmas
	3	El usuario pulsará sobre la papelera de la alarma que desee eliminar.
	5	El sistema pedirá confirmación para la eliminación.
	6	El usuario acepta la eliminación.
	4	El sistema eliminará la alarma.
Secuencia alternativa	Paso	Acción
Postcondición	La alarma quedará eliminada en el sistema.	
Excepciones	Paso	Acción
	6	El usuario descarta la eliminación.
	7	El sistema retorna a la pantalla del listado.
Rendimiento		
Importancia	Media	
Prioridad	Media	
Comentarios		

Tabla 32.- CU-10: Eliminar alarma.

CU-11	Crear clase	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario crear clases.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	Usuario selecciona la opción de menú "crear clase".
	2	El sistema muestra pantalla para introducir los datos de la clase.
	3	El usuario introduce los datos de la clase y selecciona la opción añadir horario.
	4	El sistema abre un diálogo donde permite al usuario seleccionar el día y la hora en que dará la clase.
	5	El usuario indica el día y la hora en que tendrá la clase y registra los datos
	6	El sistema crea la clase y la sitúa en la franja del horario que corresponde y lo devuelve a la pantalla desde que seleccionó la opción de menú.
Secuencia alternativa	Paso	Acción
Postcondición	La clase quedará almacenada en el sistema.	
Excepciones	Paso	Acción
	1	El usuario no completa los datos correctamente
	2	El sistema muestra un mensaje y permanece en la pantalla de creación del alumno
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 33.- CU-11: Crear clase.

CU-12	Importar clases	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario importar clases.	
Precondiciones	Inicio de sesión abierta por el usuario. Fichero situado en la carpeta importaciones dentro de la aplicación, el fichero deberá llamarse "clases".	
Secuencia Normal	Paso	Acción
	1	El usuario seleccionará la opción del menú "importar clases".
	2	El sistema almacenara las clases en la base de datos.
Secuencia alternativa	Paso	Acción
Postcondición	Las clases quedan almacenadas en el sistema	
Excepciones	Paso	Acción
	1	El usuario no completa los datos correctamente
	2	El sistema no guardará las clases
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios	El usuario deberá borrar el fichero de la carpeta tras realizar la importación.	

Tabla 34.- CU-12: Importar clase.

CU-13	Visualizar clase	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario ver las clases que imparte.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario pulsará el icono del libro situado en el menú superior de la aplicación.
	2	El sistema mostrará la pantalla que contiene todas las clases
	3	El usuario selecciona la clase que quiere ver
	4	El sistema muestra la clase seleccionada
Secuencia alternativa	Paso	Acción
	1	El usuario selecciona una clase del horario, de la pantalla principal de la aplicación.
	2	El sistema mostrará la pantalla que contiene todas las clases
	3	El usuario selecciona la clase que quiere ver
	4	El sistema muestra la clase seleccionada
Postcondición	La clase será mostrada en el sistema.	
Excepciones	Paso	Acción
	1	Si no hay clases creadas la pantalla estará vacía
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 35.- CU-13: Visualizar clase.

CU-14 Editar clase		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario editar las clases que imparte.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario pulsará el icono del libro situado en el menú superior de la aplicación.
	2	El sistema mostrará la pantalla que contiene todas las clases
	3	El usuario selecciona la clase que quiere ver
	4	El sistema muestra la clase seleccionada
	5	El usuario pulsa el icono "Edición" de la ficha de la clase
	6	El sistema muestra la pantalla de edición de la clase
	7	El usuario realiza los cambios que desee y guarda
	8	El sistema actualiza los datos de la clase.
Secuencia alternativa	Paso	Acción
	1	El usuario selecciona una clase del horario
	2	El sistema mostrará la pantalla que contiene todas las clases
	3	El usuario selecciona la clase que quiere ver
	4	El sistema muestra la clase seleccionada
	5	El usuario pulsa el icono "Edición" de la ficha de la clase
	6	El sistema muestra la pantalla de edición de la clase
	7	El usuario realiza los cambios que desee y guarda
	8	El sistema actualiza los datos de la clase.
Postcondición	La clase quedara actualizada en el sistema.	
Excepciones	Paso	Acción
	1	Si no hay clases creadas la pantalla estará vacía.
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 36.- CU-14: Editar clase.

CU-15		Crear alumno	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario crear alumnos.		
Precondiciones	Inicio de sesión abierta por el usuario.		
Secuencia Normal	Paso	Acción	
	1	El usuario seleccionará la opción del menú “crear alumno”.	
	2	El sistema mostrará la pantalla de creación del alumno, que contendrá todos los datos necesarios para crear la ficha del alumno.	
	3	El usuario completa los campos y presiona el botón “registrar”	
	4	El sistema registra el alumno y lo sitúa dentro de la clase que le corresponda.	
Secuencia alternativa	Paso	Acción	
Postcondición	El alumno queda almacenado en el sistema		
Excepciones	Paso	Acción	
	3	El usuario no completa los datos correctamente	
	4	El sistema muestra un mensaje y permanece en la pantalla de creación del alumno	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 37.- CU-15: Crear alumno.

CU-16		Importar alumnos	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario importar alumnos.		
Precondiciones	Inicio de sesión abierta por el usuario. Fichero situado en la carpeta importaciones de la aplicación.		
Secuencia Normal	Paso	Acción	
	1	El usuario seleccionará la opción del menú “importar alumnos”.	
	2	El sistema almacenará los alumnos en la base de datos.	
Secuencia alternativa	Paso	Acción	
Postcondición	Los alumnos quedan almacenados en el sistema		
Excepciones	Paso	Acción	
	2	El usuario no completa los datos correctamente	
	3	El sistema no guardara los alumnos	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 38.- CU-16: Importar alumno.

UC-17	Visualizar ficha alumno	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario ver la ficha de un alumno.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario pulsará el icono del libro situado en el menú superior de la aplicación.
	2	El sistema mostrará la pantalla que contiene todas las clases
	3	El usuario seleccionará una clase.
	4	El sistema retornará la pantalla con los alumnos pertenecientes a esa clase.
	5	El usuario seleccionará un alumno
	6	El sistema mostrará la pantalla que contiene la ficha del alumno seleccionado.
Secuencia alternativa	Paso	Acción
	1	El usuario pulsará la clase que desee de la pantalla del horario.
	2	El sistema retornará la pantalla con los alumnos pertenecientes a esa clase.
	3	El usuario seleccionará un alumno
	4	El sistema mostrará la pantalla que contiene la ficha del alumno seleccionado.
Postcondición	El alumno queda almacenado en el sistema	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios	Este caso de uso completa los requisitos de visualización de anotaciones y objetivos	

Tabla 39.- CU-17: Visualizar alumno.

UC-18		Editar alumno	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario editar un alumno.		
Precondiciones	Inicio de sesión abierta por el usuario.		
Secuencia Normal	Paso	Acción	
	1	El usuario pulsará el icono del libro situado en el menú superior de la aplicación.	
	2	El sistema mostrará la pantalla que contiene todas las clases	
	3	El usuario seleccionará una clase.	
	4	El sistema retornará la pantalla con los alumnos pertenecientes a esa clase.	
	5	El usuario seleccionará un alumno	
	6	El sistema mostrará la pantalla que contiene la ficha del alumno seleccionado.	
	7	El usuario pulsa el botón “edición” de la ficha del alumno.	
	8	El sistema muestra la pantalla de edición del alumno.	
	9	El usuario modifica los campos deseados y pulsa sobre el botón aceptar.	
	10	El sistema actualiza los datos del alumno.	
Secuencia alternativa	Paso	Acción	
	1	El usuario pulsará la clase que desee de la pantalla del horario.	
	2	El sistema retornará la pantalla con los alumnos pertenecientes a esa clase.	
	3	El usuario seleccionará un alumno	
	4	El sistema mostrará la pantalla que contiene la ficha del alumno seleccionado.	
	5	El usuario pulsa el botón “edición” de la ficha del alumno.	
	6	El sistema muestra la pantalla de edición del alumno.	
	7	El usuario modifica los campos deseados y pulsa sobre el botón aceptar.	
	8	El sistema actualiza los datos del alumno.	
Postcondición	El alumno queda actualizado en el sistema		
Excepciones	Paso	Acción	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 40.- CU-18: Editar alumno.

UC-19	Eliminar alumno	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario eliminar a un alumno.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario pulsará el icono del libro situado en el menú superior de la aplicación.
	2	El sistema mostrará la pantalla que contiene todas las clases
	3	El usuario seleccionará una clase.
	4	El sistema retornará la pantalla con los alumnos pertenecientes a esa clase.
	5	El usuario pulsará el icono “papelera” del alumno que desee borrar.
	6	El sistema eliminará al alumno y actualizará la pantalla con la lista de alumnos.
Secuencia alternativa	Paso	Acción
	1	El usuario pulsará la clase que desee de la pantalla del horario.
	2	El sistema retornará la pantalla con los alumnos pertenecientes a esa clase.
	3	El usuario pulsará el icono “papelera” del alumno que desee borrar.
	4	El sistema eliminará al alumno y actualizará la pantalla con la lista de alumnos.
Postcondición	El alumno queda eliminado del sistema	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 41.- CU-19: Eliminar alumno.

CU-20	Crear anotación	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario crear anotaciones sobre los alumnos.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsará el icono de grabación.
	2	El sistema devolverá un pop-up para la grabación
	3	El usuario pulsará sobre la grabación y dirá lo que quiere transcribir.
	4	El sistema escribirá en el pop-up la anotación realizada por el usuario.
	5	El usuario acepta lo transcrito
	6	El sistema guarda la anotación
Secuencia alternativa	Paso	Acción
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsará el botón "Añadir nota".
	2	El sistema devolverá un pop-up para la grabación.
	3	El usuario escribirá lo que quiere y pulsará el botón aceptar.
	4	El sistema guarda la anotación
Postcondición	La anotación queda almacenada en el sistema	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 42.- CU-20: Crear anotación.

CU-21 Editar anotación		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario editar comentarios sobre los alumnos.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsará el icono de edición del comentario.
	2	El sistema devolverá un pop-up para la edición.
	3	El usuario cambiará el comentario y pulsará aceptar.
	4	El sistema actualiza el comentario.
Secuencia alternativa	Paso	Acción
Postcondición	La anotación queda actualizada en el sistema	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 43.- CU-21: Editar anotación.

CU-22 Eliminar anotación		
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario eliminar anotaciones sobre los alumnos.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsará el icono “papelera” de la anotación.
	2	El sistema preguntará si desea eliminar la anotación
	3	El usuario acepta la eliminación.
	4	El sistema elimina la anotación.
Secuencia alternativa	Paso	Acción
Postcondición	La anotación queda eliminada del sistema.	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 44 .- CU-22: Eliminar anotación.

CU-23		Visualizar horario	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario ver la planificación de su horario de clases.		
Precondiciones	Inicio de sesión abierta por el usuario.		
Secuencia Normal	Paso	Acción	
	1	Aparecerá tras el inicio de sesión.	
Secuencia alternativa	Paso	Acción	
Postcondición	El sistema mostrara la pantalla del horario		
Excepciones	Paso	Acción	
	1	Se podrá retornar dando el botón “atrás” del teléfono móvil.	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 45.- CU-23: Visualizar horario.

CU-24		Crear objetivo	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario crear objetivos.		
Precondiciones	Inicio de sesión abierta por el usuario.		
Secuencia Normal	Paso	Acción	
	1	El usuario seleccionará la opción del menú “crear objetivo”.	
	2	El sistema mostrará la pantalla de creación del objetivo, que contendrá todos los datos necesarios para crear el objetivo.	
	3	El usuario completa los campos y presiona el botón “aceptar”	
	4	El sistema registra el objetivo.	
Secuencia alternativa	Paso	Acción	
Postcondición	El objetivo queda almacenado en el sistema		
Excepciones	Paso	Acción	
	1	El usuario no completa los datos correctamente	
	2	El sistema muestra un mensaje y permanece en la pantalla de creación del objetivo.	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 46.- CU-24: Crear objetivo.

CU-25	Importar objetivo	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario importar objetivos.	
Precondiciones	Inicio de sesión abierta por el usuario. Fichero situado en la carpeta importaciones dentro de la aplicación.	
Secuencia Normal	Paso	Acción
	1	El usuario seleccionará la opción del menú "importar objetivos".
	2	El sistema almacenará los objetivos en la base de datos.
Secuencia alternativa	Paso	Acción
Postcondición	Los objetivos quedan almacenados en el sistema	
Excepciones	Paso	Acción
	1	El usuario no completa los datos correctamente
	2	El sistema no guardará los alumnos
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 47.- CU-25: Importar objetivo

CU-26	Asignar objetivo	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario asociar los objetivos sobre los alumnos.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsará el icono de objetivo.
	2	El sistema devolverá una pantalla con los objetivos pendientes de asignar.
	3	El usuario seleccionará al objetivo.
	4	El sistema asociará el objetivo al alumno en el que se encuentra.
Secuencia alternativa	Paso	Acción
	1	El usuario se desplazará a la pantalla de la clase y pulsará el icono de objetivo.
	2	El sistema mostrará una pantalla con la lista de objetivos pendientes de asignar.
	3	El usuario seleccionará el objetivo deseado y aceptará.
	4	El sistema asociará el objetivo a los alumnos pertenecientes a la clase en la que se encuentra situado.
Postcondición	El objetivo quedara asignado a uno o varios alumnos.	
Excepciones	Paso	Acción
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 48.- CU-26: Asignar objetivo.

CU-27		Aprobar objetivo	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario aprobar los objetivos sobre los alumnos.		
Precondiciones	Inicio de sesión abierta por el usuario.		
Secuencia Normal	Paso	Acción	
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsará el deslizable para la aprobación del objetivo.	
	2	El sistema actualizará el objetivo.	
Secuencia alternativa			
Postcondición	El objetivo quedara aprobado en el alumno.		
Excepciones	Paso	Acción	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 49.- CU-27: Aprobar objetivo.

CU-28		Eliminar objetivo	
Versión	1.0 (01/01/2018)		
Autores	Clara Herranz de Andrés		
Descripción	El sistema deberá permitir al usuario eliminar los objetivos sobre los alumnos.		
Precondiciones	Inicio de sesión abierta por el usuario.		
Secuencia Normal	Paso	Acción	
	1	El usuario se desplazará a la pantalla de la ficha del alumno y pulsara el botón "papelera" del objetivo.	
	2	El sistema preguntara si se desea borrar el objetivo del alumno.	
	3	El usuario aceptara el borrado	
	4	El sistema borrara el objetivo	
Secuencia alternativa			
Postcondición	El objetivo quedará eliminado.		
Excepciones	Paso	Acción	
	3	El usuario cancela la eliminación del objetivo.	
	4	El sistema no borra el objetivo.	
Rendimiento			
Importancia	Alta		
Prioridad	Alta		
Comentarios			

Tabla 50.- CU-28: Eliminar objetivo.

CU-29	Borrar datos	
Versión	1.0 (01/01/2018)	
Autores	Clara Herranz de Andrés	
Descripción	El sistema deberá permitir al usuario, tras finalizar el curso, eliminar todos los datos para comenzar un nuevo uso.	
Precondiciones	Inicio de sesión abierta por el usuario.	
Secuencia Normal	Paso	Acción
	1	El usuario seleccionará del menú la opción “borrar datos”
	2	El sistema pregunta al usuario si quiere eliminar todos los datos de la aplicación
	3	El usuario acepta la eliminación.
	4	El sistema eliminará, menos los datos del docente, todos los datos almacenados tanto en la base de datos como en las preferencias compartidas y volverá a la pantalla de configuración del horario.
Secuencia alternativa	Paso	Acción
Postcondición	Los datos serán eliminados en el sistema y la sesión será cerrada.	
Excepciones	Paso	Acción
	3	El usuario cancela la acción.
	4	El sistema vuelve a la pantalla anterior sin realizar ninguna acción.
Rendimiento		
Importancia	Alta	
Prioridad	Alta	
Comentarios		

Tabla 51.- CU-29: Borrar datos.

4.3. Requisitos no funcionales

RNF-01: La aplicación móvil deberá ser ejecutable desde cualquier dispositivo con sistema operativo Android, con versión mínima 4.0.

RNF-02: La aplicación móvil deberá estar operativa 24 horas al día los 7 días de la semana.

RNF-03: Aunque la aplicación sólo podrá soportar un usuario por dispositivo, por seguridad dispondrá de autenticación mediante usuario y contraseña.

RNF-04: Las fechas tendrán el formato “dd/mm/yyyy” y las horas” HH:mm”.

RNF-05: El sistema debe ser escalable con el fin de que las funcionalidades conserven su calidad cuando el sistema aumente.

RNF-06: El sistema no debe corromper los datos utilizados en la aplicación.

RNF-07: El sistema deberá tener un diseño sencillo y funcional, debido a que los conocimientos de usuarios a la que se destina son muy diversos entre sí.

RNF-08: La primera versión solo soportará el idioma español.

4.4. Requisitos de información

RI-01: El email del usuario deberá tener el siguiente patrón ****@****.***

RI-02: El sistema almacenará los datos del alumno (nombre, apellidos, fecha de nacimiento, clase).

RI-03: La foto de perfil del alumno no será obligatoria.

RI-04: El sistema almacenará los datos de la clase (curso, grupo).

RI-06: El sistema almacenará los datos del objetivo (nombre, descripción, fecha).

RI-06: El sistema almacenará los datos de las alarmas (título, fecha, hora repetir).

RI-07: El sistema almacenará los datos del usuario (usuario, contraseña, email) en las preferencias compartidas.

RI-08: El sistema almacenará las franjas horarias en las preferencias compartidas.

RI-09: El sistema almacenará los datos de la anotación (fecha, nota, alumno).

RI-10: El sistema almacenará los datos de los horarios de las clases (posición, día, horalnicio, minutoInicio, horaFin, minutoFin, clase).

4.5. Diseño conceptual

En este apartado veremos el modelo conceptual, que representa las relaciones que surgen entre las entidades vistas anteriormente. Este modelo también es conocido como “Entidad-Relación”.

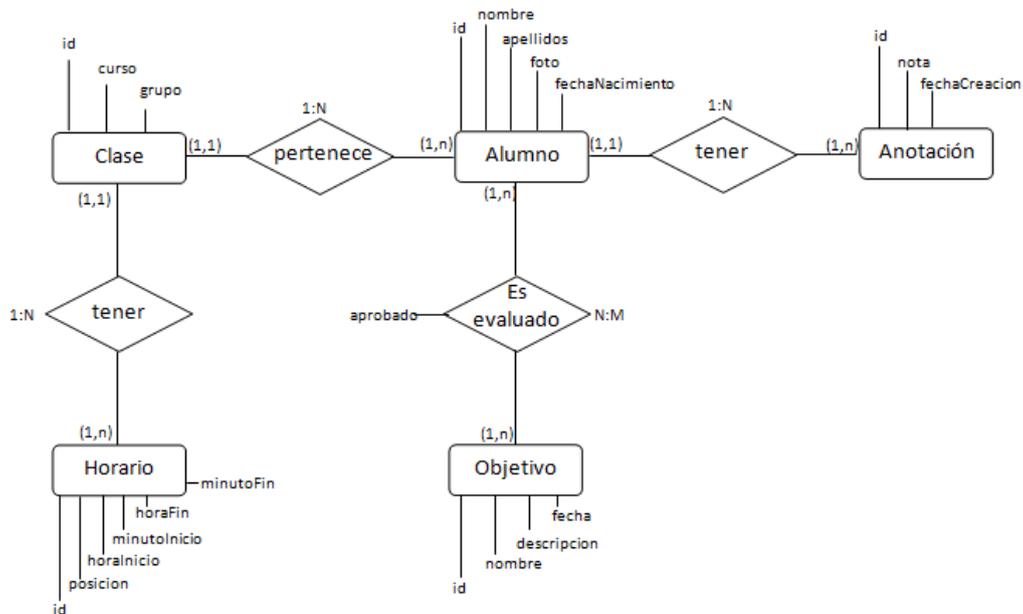


Figura 8.- Modelo Entidad-Relación BBDD Clase.

El sistema posee dos bases de datos, la primera contiene todo lo relacionado al tratamiento de las clases y evaluación de alumnos, su entidad principal es Alumno. Posee un conjunto de atributos (id, nombre, apellidos, foto, fechaNacimiento) que forman parte de su ficha de información.

Esta entidad a su vez se relaciona con otras tres que terminan de completar la información sobre el alumno:

- La entidad **Clase** con la que tiene una relación 1:N, ya que un alumno únicamente puede tener una clase, sin embargo una clase puede tener varios alumnos.
- La entidad **Anotación** con la cual posee una relación 1:N también, ya que un alumno puede tener varias anotaciones, sin embargo una anotación solo puede pertenecer a un alumno.
- Por último también existe una relación con la entidad **Objetivos** con la cual posee una relación N:M, ya que un alumno puede tener varios objetivos y un objetivo puede pertenecer a varios alumnos.

Por otro lado la entidad Clase posee relación con la entidad Horario. Esta relación es 1:N, ya que una clase puede tener varios horarios pero un horario solo tendrá una clase. Esta relación servirá para representar el horario de la clase.

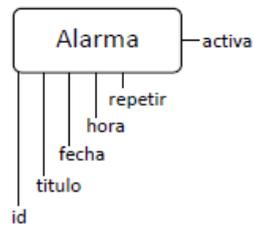


Figura 9.- Modelo Entidad-Relación BBDD Alarma.

Por último, tenemos la base de datos que gestiona las alarmas, una alarma puede estar programada por cualquier motivo, no tiene por qué tener relación con una clase concreta o un alumno determinado, por lo que esta base de datos únicamente contiene la entidad Alarma.

Capítulo 5 - Diseño de Software

En este apartado, se describirán los componentes y la arquitectura tanto lógica como física de la aplicación móvil.

5.1. Arquitectura lógica

En la arquitectura lógica podemos ver los componentes lógicos que forman el sistema así como su relación. Se dividirá la arquitectura en dos puntos: por un lado se mostrará la arquitectura lógica del sistema completo, donde definiremos sus principales características y porqué se ha decidido implementarla, y posteriormente la de la aplicación en la que detallaremos los componentes que forman cada parte de la arquitectura en nuestro sistema.

5.1.1. Arquitectura lógica del sistema

La arquitectura del sistema sigue un modelo en tres capas:

- **La capa de presentación:** contiene los componentes que forman la interfaz de usuario. Su función principal es transmitir las acciones que realice el usuario a la capa de negocio.
- **La capa de negocio:** Esta capa se encarga de la lógica de la aplicación. Realiza todas las operaciones, tanto las que comunican al usuario con el sistema como las que comunican el sistema con la base de datos.

- **La capa de datos:** Esta es la capa encargada de almacenar la información. En ella se gestionará todo lo relacionado con la base de datos (creación, borrado y edición de datos).

Se ha elegido esta arquitectura debido a que cada capa únicamente se comunica con la inmediatamente inferior, de tal manera, que si se debe realizar algún cambio, no será necesario modificar el sistema al completo, sólo la parte afectada.

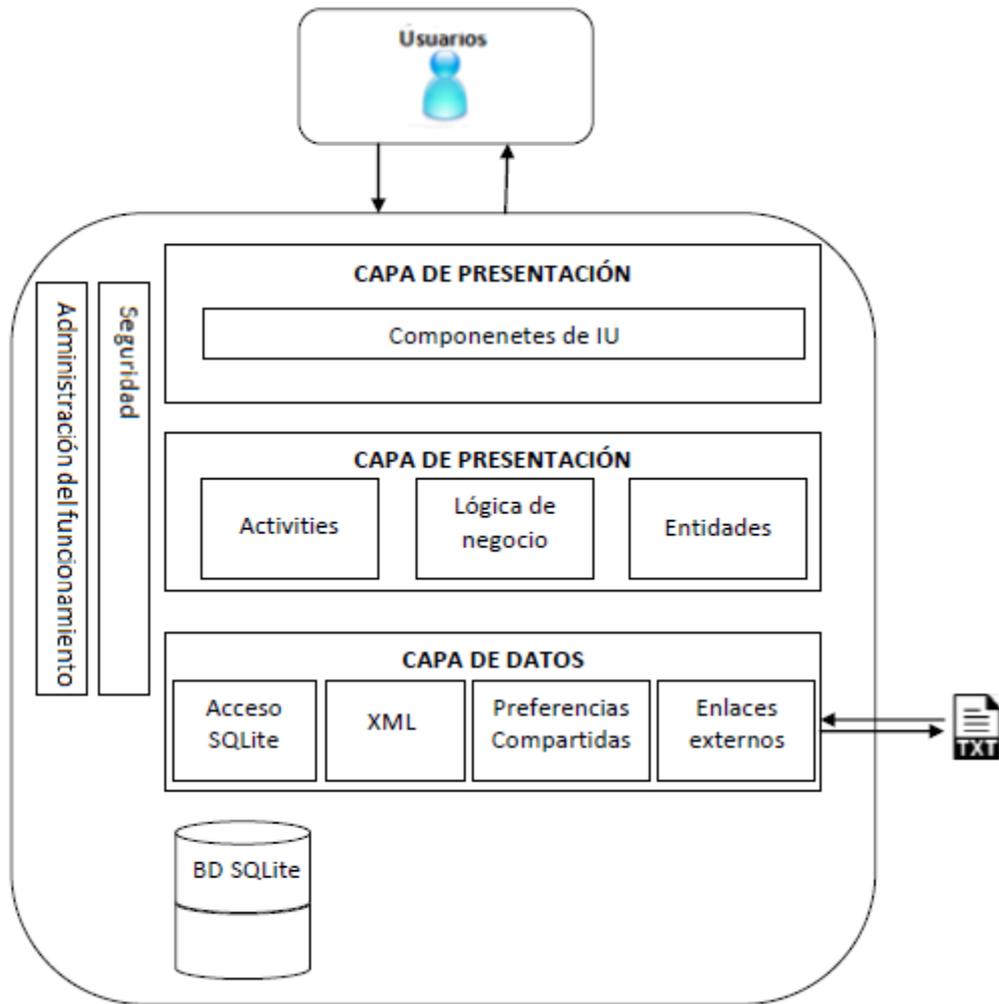


Figura 10.- Arquitectura lógica del sistema.

5.1.2. Arquitectura lógica de la aplicación

La arquitectura lógica de la aplicación se compone de tres capas como se ha visto en el apartado anterior. La capa de presentación está formada por todas las pantallas (activities xml) que componen la aplicación.

La capa de negocio se compone de todas las clases java, contienen las funcionalidades de la aplicación y se relacionan con la capa de presentación (para la recogida de datos de formularios, dibujar horario etc) y con la capa de datos (para la gestión de la información del sistema).

La capa de datos gestiona la información del sistema. La base de datos SQLite almacena toda la información que interviene en el sistema, salvo los datos del usuario y las franjas horarias, que son almacenadas en las preferencias compartidas.

Los ficheros txt de importación y exportación tienen relación con el sistema al completo, ya que su ejecución se lanza desde la capa de presentación, la información se trata en la capa de negocio y la capa de datos almacena/provee esos datos.

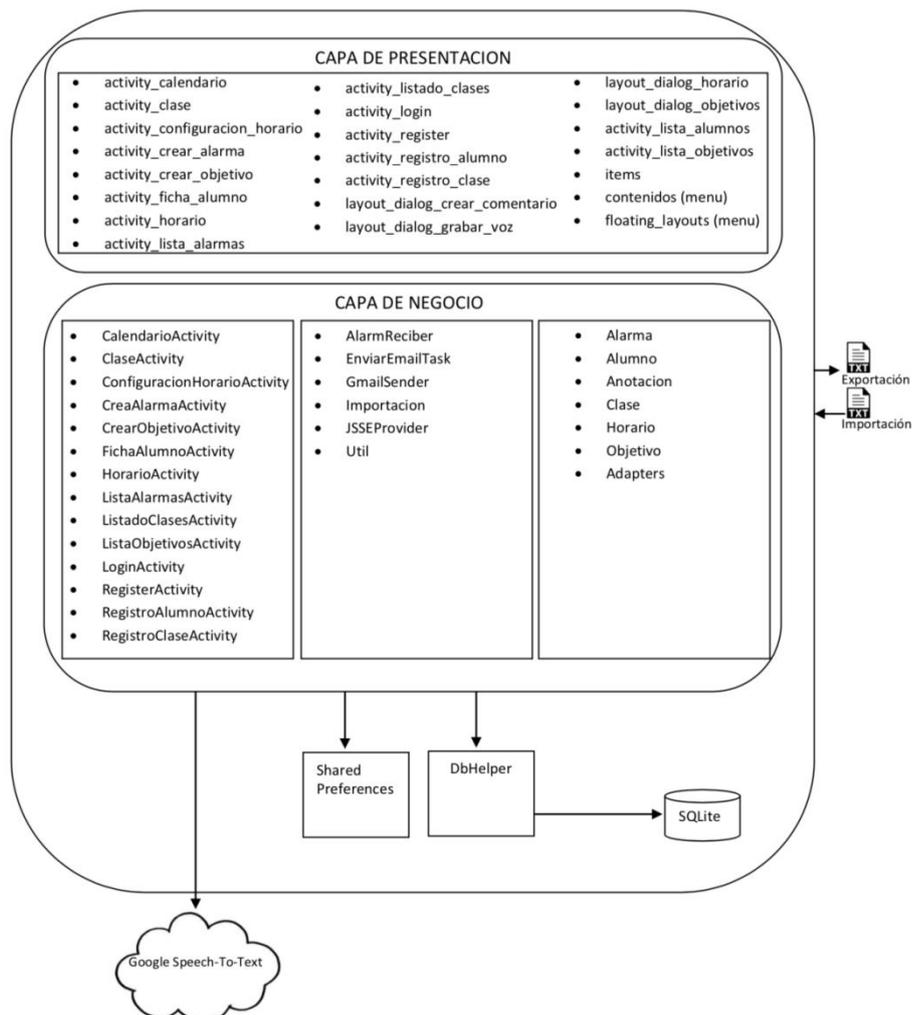


Figura 11.- Arquitectura lógica de la aplicación.

La estructura de la aplicación se resume de la siguiente forma:

- **Aplicación: Componentes de la aplicación.**
 - **Capa de Negocio:** componentes que definen el comportamiento de la aplicación, veremos los que se asocian directamente con la interfaz de usuario:
 - **CalendarioActivity:** pantalla del calendario (activity_calendario).
 - **ClaseActivity:** pantalla de una clase (activity_clase).
 - **ListaAlumnosAdapter:** contiene el conjunto de los alumnos.
 - **ConfiguracionHorarioActivity:** pantalla de configuración del horario, aparecerá tras el primer inicio de sesión del usuario (activity_configuracion_horario).
 - **CrearAlarmaActivity:** pantalla de creación de la alarma (activity_crear_alarma).
 - **AlarmReciber:** cuando se crea la alarma, se crea el evento que se lanzará en su vencimiento.
 - **CrearObjetivoActivity:** pantalla de creación de objetivos (activity_crear_objetivo).
 - **FichaAlumnoActivity:** pantalla del alumno (activity_ficha_alumno).
 - **ListaAnotacionesAdapter:** contiene la lista de anotaciones del alumno.
 - **ListaObjetivosAdapter:** contiene la lista de objetivos del alumno.
 - **HorarioActivity:** pantalla principal del sistema. Contiene el horario (activity_horario).
 - **ListaAlarmasActivity:** pantalla que contiene las alarmas (activity_lista_alarmas).
 - **ListaAlarmasAdarter:** contiene la lista de alarmas de la aplicación.
 - **ListadoClasesActivity:** pantalla de clases (activity_listado_clases).
 - **ListaClasesAdapter:** contiene el listado de las clases.
 - **LoginActivity:** pantalla de inicio de sesión (activity_login).
 - **GmailSender:** usado para el envío del correo.
 - **EnviarMailTask:** usado para el envío del correo.
 - **JSSEProvider:** usado para el envío del correo.
 - **RegisterActivity:** pantalla de registro del usuario (activity_register).
 - **RegistroAlumnoActivity:** pantalla de creación de un alumno (activity_registro_alumno).
 - **RegistroClaseActivity:** pantalla de creación de la clase (activity_registro_clase).
 - **DbHelper:** conexión con la base de datos.
 - **APIs:** Interfaces de aplicaciones de Google usadas.
 - **Google Speech-To-Text:** API para la transcripción de audio en texto.

5.2. Arquitectura física

En la arquitectura física podemos ver como el sistema está compuesto por la base de datos SQLite, el cliente y el uso de los servicios de Google.

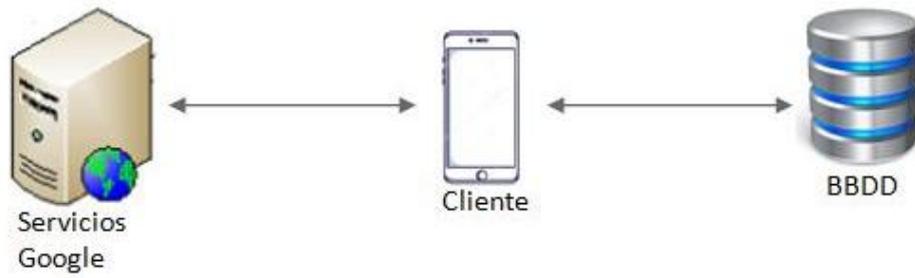


Figura 12.- Arquitectura física.

5.3. Diagrama de clases

Debido a la gran cantidad de clases que forman la aplicación móvil, se crearán diferentes diagramas de clases. En el primero de ellos se podrán observar los paquetes en los que se divide la aplicación, y posteriormente se definirá un diagrama de clases por cada paquete.

La estructura de clases de la aplicación se ha dividido en 5 paquetes:

- **SQLite:** contiene las clases de estructura y métodos relativos a la base de datos.
- **Modelo:** se encuentran las clases que representan las entidades almacenadas en la base de datos.
- **Activity:** contiene las clases java que desarrollan las funcionalidades directas de los ficheros XML e implementan los métodos invocados por las actividades a los que representan.
- **Util:** contiene funciones desarrolladas que se utilizan en los activities.
- **Listener:** implementa las opciones del menú deslizable lateral.

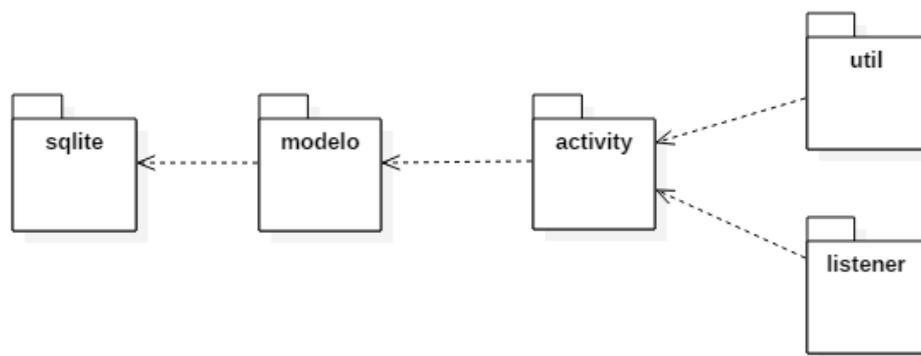


Figura 13.- Diagrama de Clases, representación de paquetes.

5.3.1. Paquete SQLite.

Este paquete está formado por dos clases .java, la primera “DBEstructura” en la que se define la estructura de la base de datos, las tablas con sus nombres y los componentes que la forman.

La clase DBHelper contiene todos los métodos de la base de datos que interactuarán con el sistema guardando los datos a través de las clases del paquete “Modelo”.

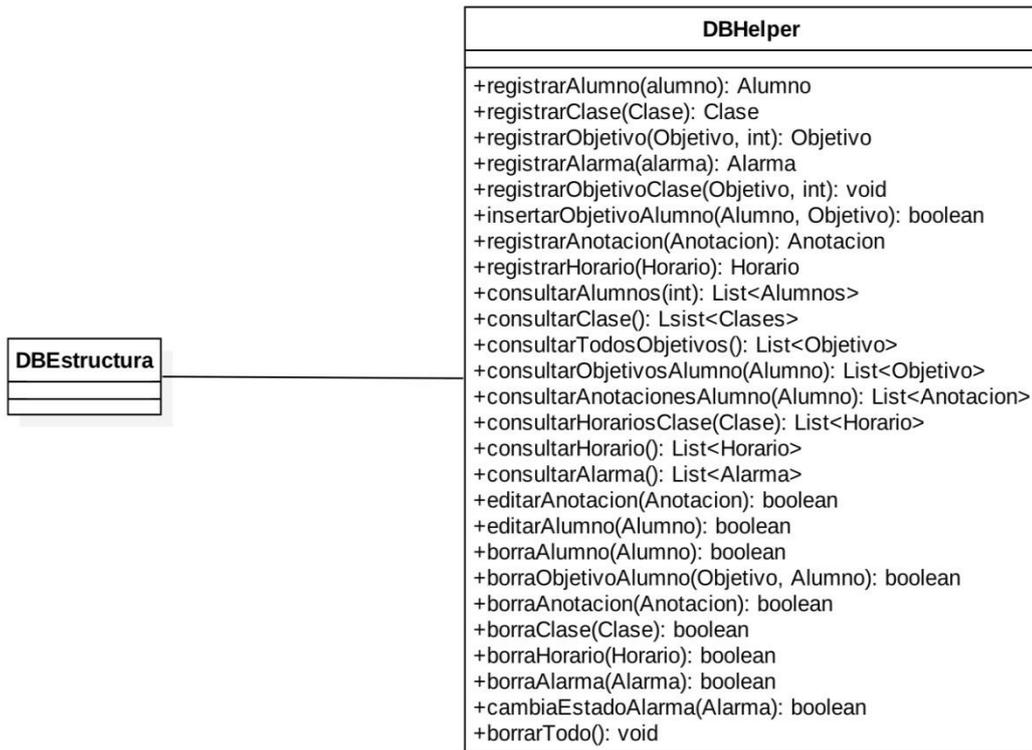


Figura 14.- Diagrama de Clases, representación del paquete SQLite.

5.3.2. Paquete modelo.

En este paquete están definidas las clases de java que formarán los objetos que serán almacenados en las dos bases de datos. Todas estas clases disponen de su método constructor y sus respectivos getters/setters, aunque no han sido representados para mejorar la visibilidad de las clases.

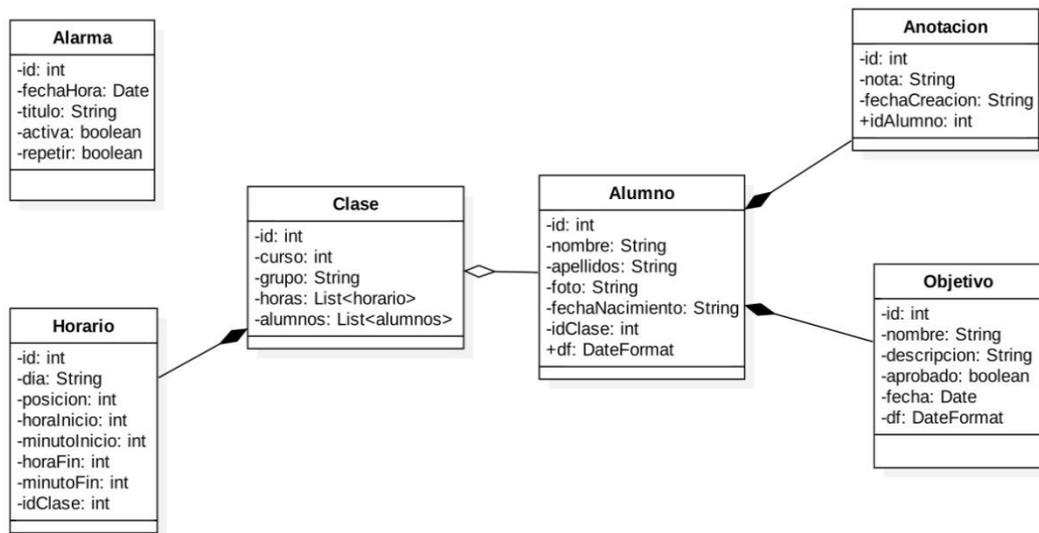


Figura 15.- Diagrama de Clases, representación del paquete modelo.

5.3.3. Paquete activity.

En este paquete están todas las clases java que interfieren directamente con la interfaz de usuario, es decir, con las activities XML. Son las responsables de la funcionalidad del sistema. Debido a la gran cantidad de métodos que poseen, por visibilidad, se han omitido métodos comunes como onCreate, onOptionsItemSelected u onActivityResult.

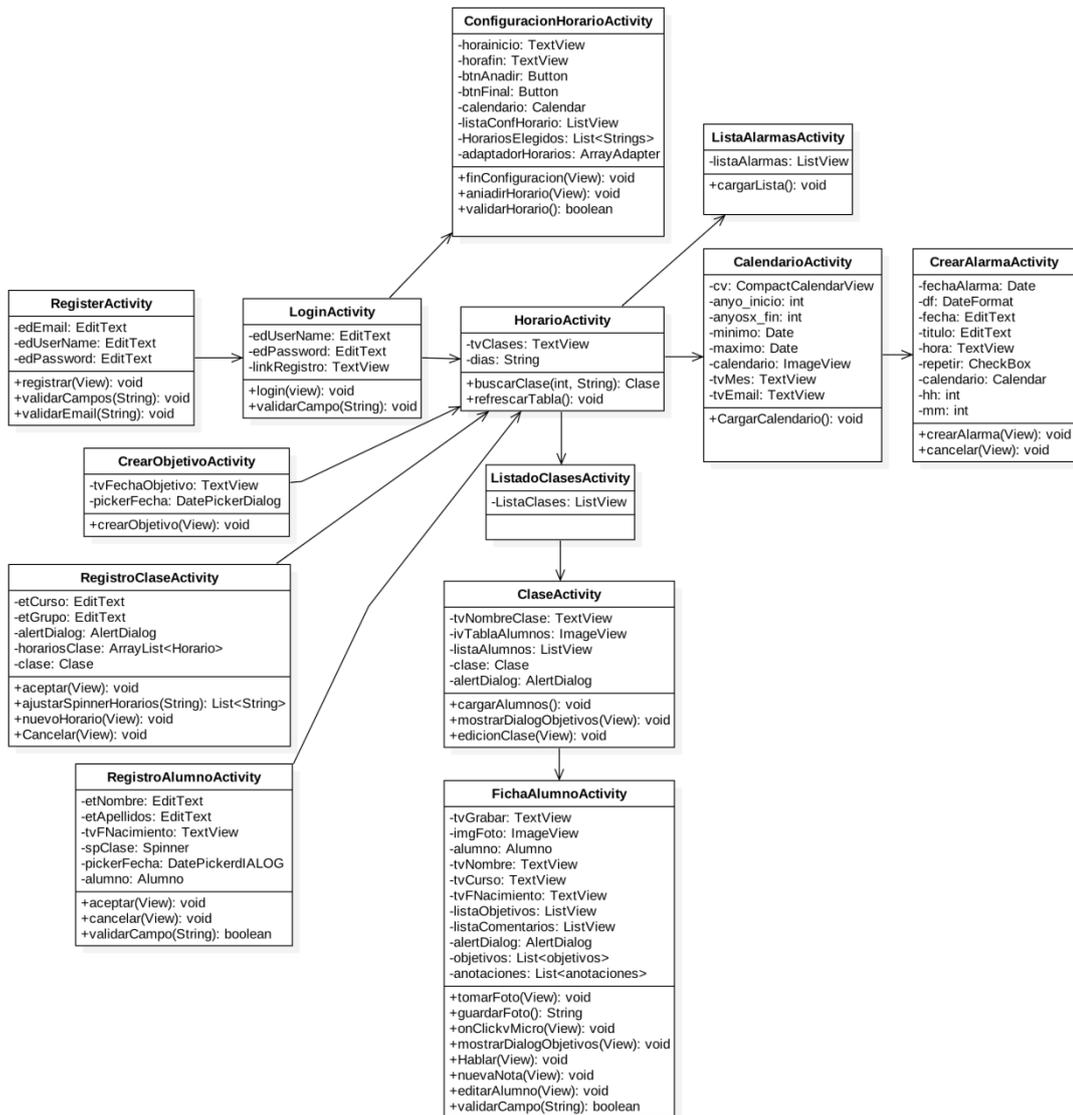


Figura 16.- Diagrama de Clases, representación del paquete activity.

5.3.4. Paquete util.

Contiene funcionalidad utilizada en toda la aplicación, que debido a su tamaño o su uso repetido, se ha decidido extraer a este paquete por ahorro de código y legibilidad de las clases.

Dispone de seis clases, tres de las cuales están destinadas al envío del email en caso de necesitar recuperar la contraseña. Estas clases son:

- **JSSEProvider**: cuya utilidad es establecer la comunicación, en ella se implementa el uso del protocolo TLS.
- **GmailSender**: define tanto las características del envío, como el puerto utilizado o el protocolo de autenticación.
- **EnviarMailTask** implementa el método de envío, definiendo el contenido que se va a enviar y el usuario que lo envía, haciendo uso de GmailSender.

La clase “AlarmReciber” se encarga de lanzar los eventos creados en el calendario. La clase “Importación” será la encargada de la importación de alumnos, clases y objetivos, y la exportación de todos los datos.

Por último, la clase “Util”, contiene funciones varias como el lanzamiento del mensaje de confirmación de borrado, el cierre de sesión, un método que localiza qué posición del horario está siendo seleccionada y un método para la gestión de la pila.

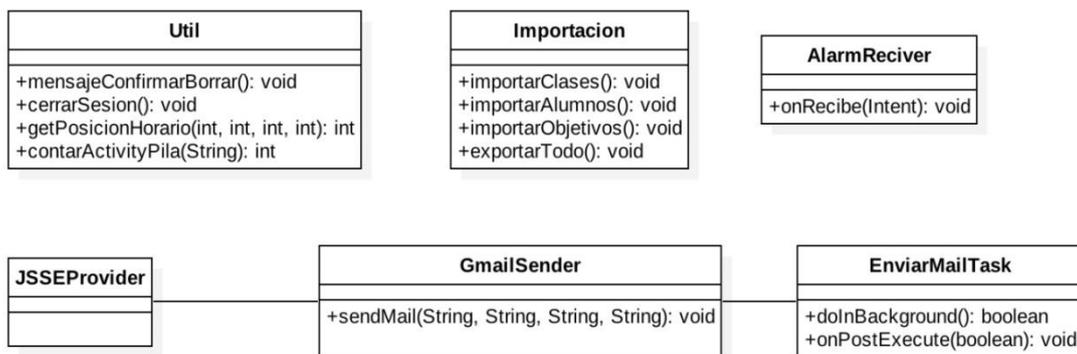


Figura 17.- Diagrama de Clases, Representación del paquete util.

5.3.5. Paquete listener.

Este paquete únicamente posee una clase, encargada de que al seleccionar una opción de menú, esta vaya al método que realiza su funcionalidad.

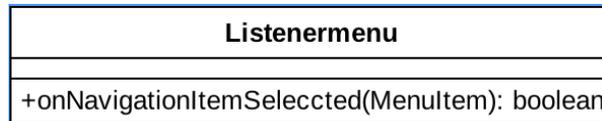


Figura 18.- Diagrama de Clases, representación del paquete listener.

5.4. Diagramas de secuencia

Los diagramas de secuencia describen las interacciones entre los componentes del sistema a lo largo del tiempo.

A continuación se mostrarán los diagramas de secuencia más significativos de la aplicación, ya que hay un gran número de diagramas que son muy similares y no se identificarían grandes variaciones entre sí (como, por ejemplo, todos los relativos a la creación de elementos, del cual se representará un único diagrama).

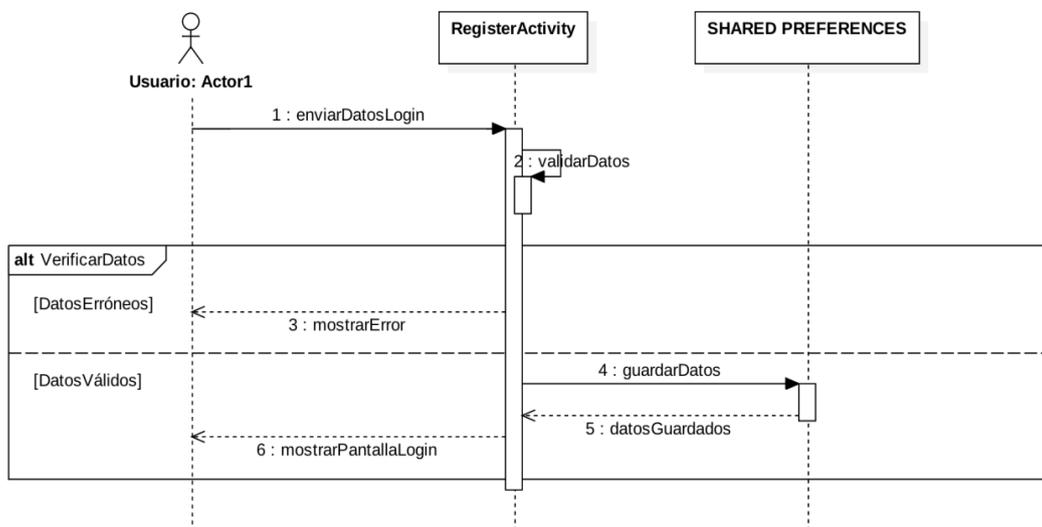


Figura 19.- Diagrama de secuencia - Registro en el sistema.

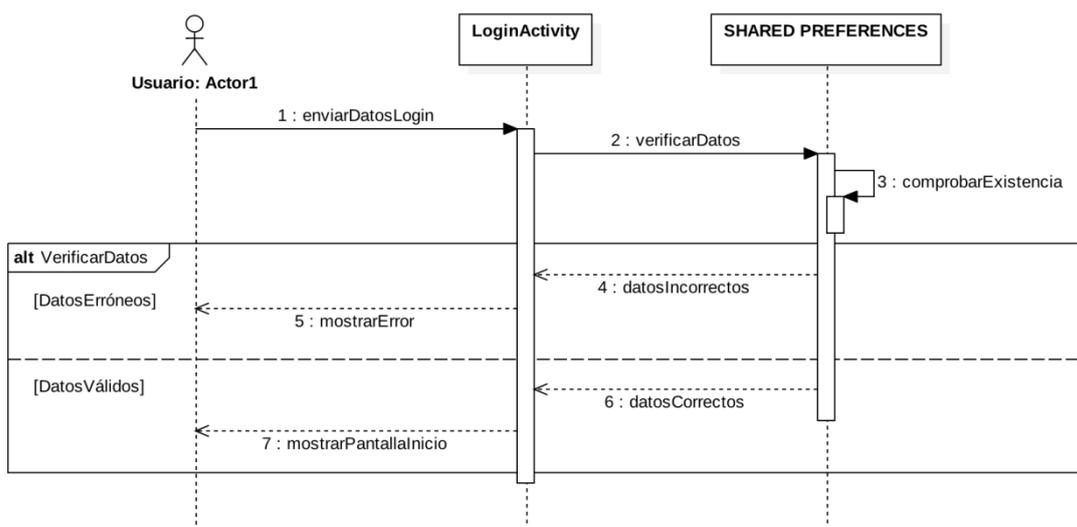


Figura 20.- Diagrama de secuencia - Login en el sistema.

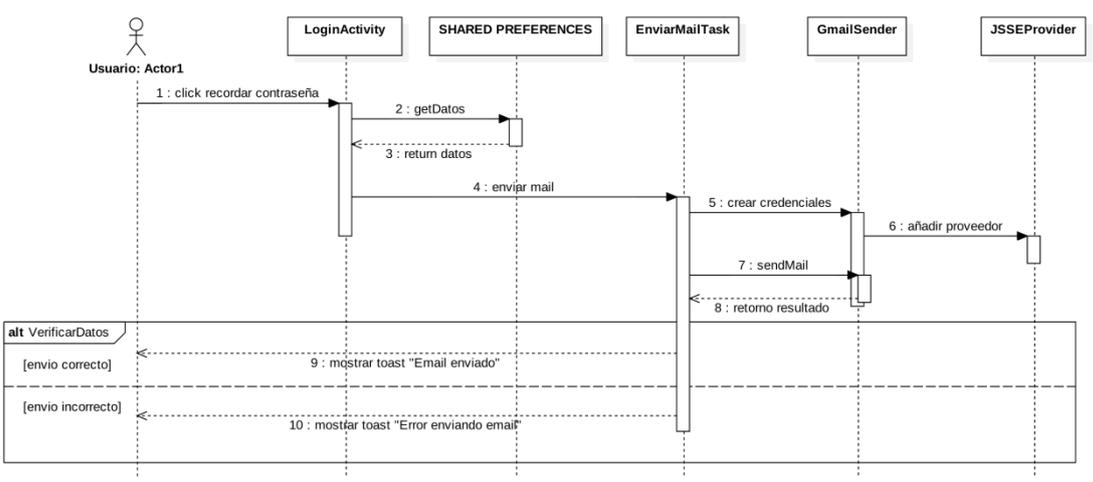


Figura 21.- Diagrama de secuencia - Recuperar contraseña.

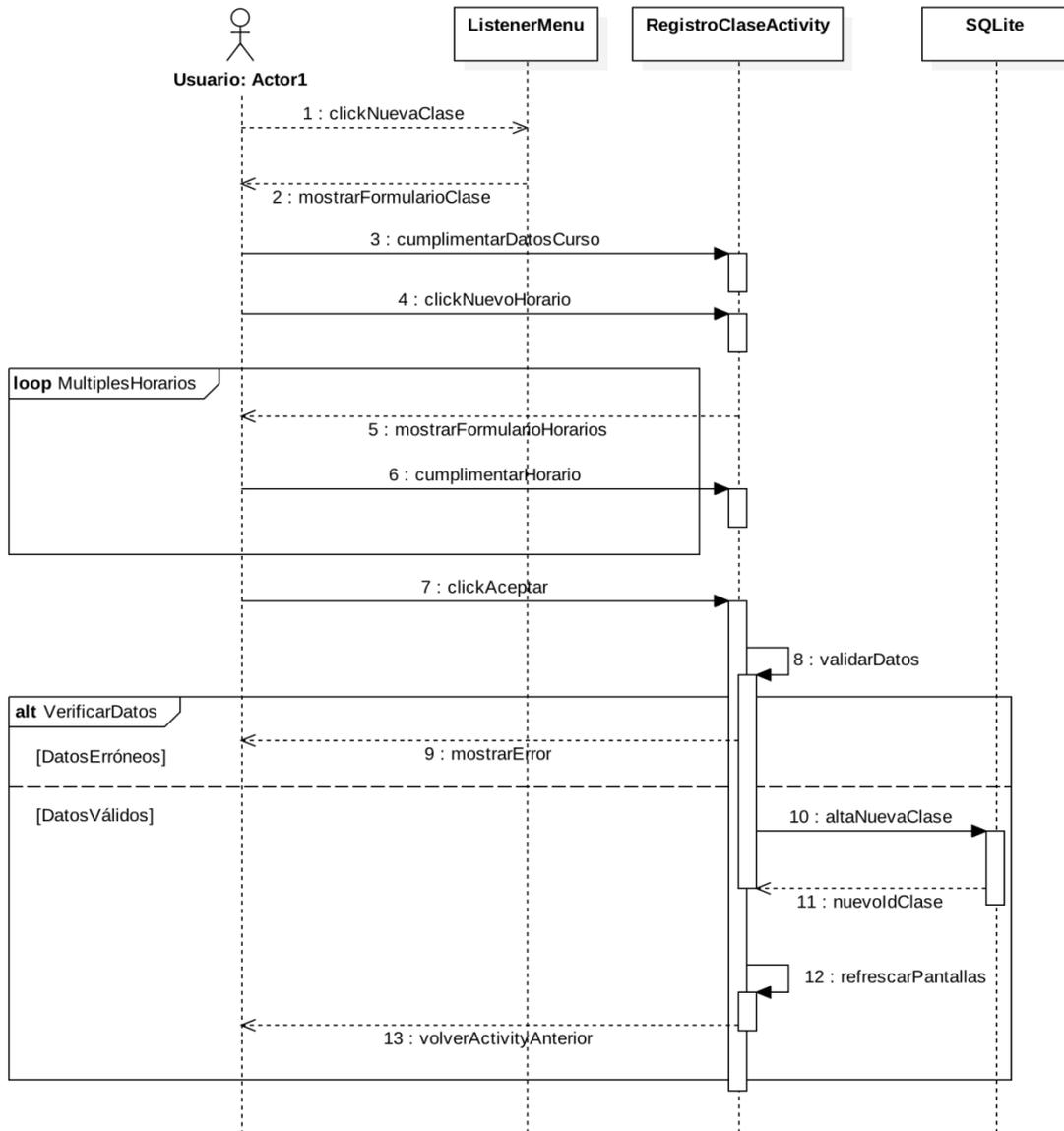


Figura 22.- Diagrama de secuencia - Crear Clase.

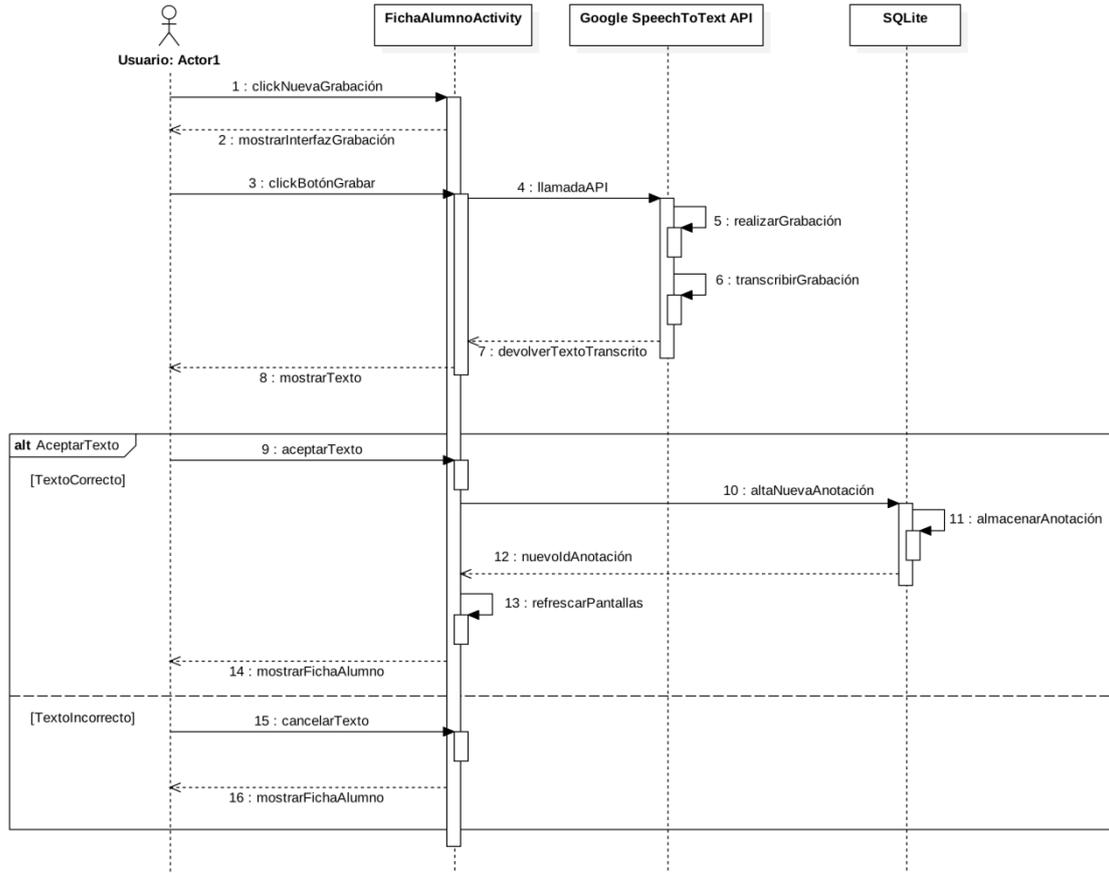


Figura 23.- Diagrama de secuencia - Crear anotación.

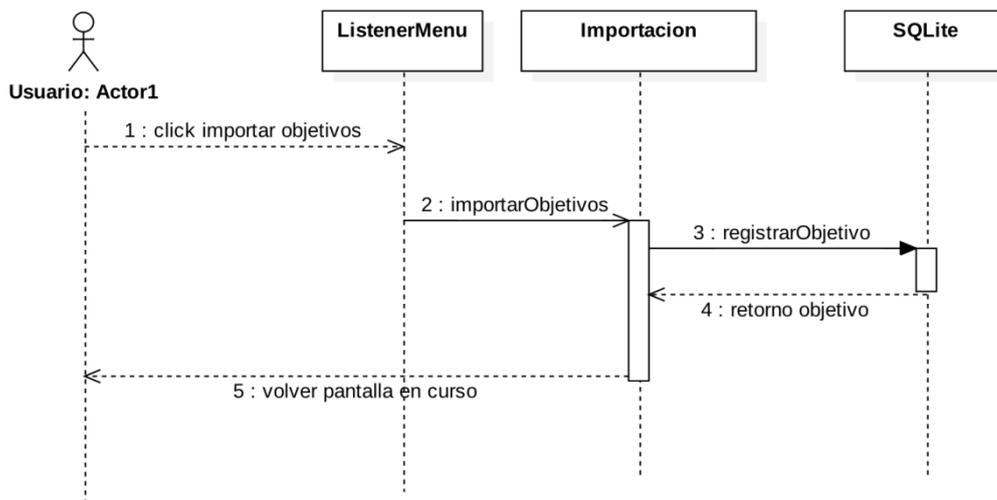


Figura 24.- Diagrama de secuencia - Importar Objetivo.

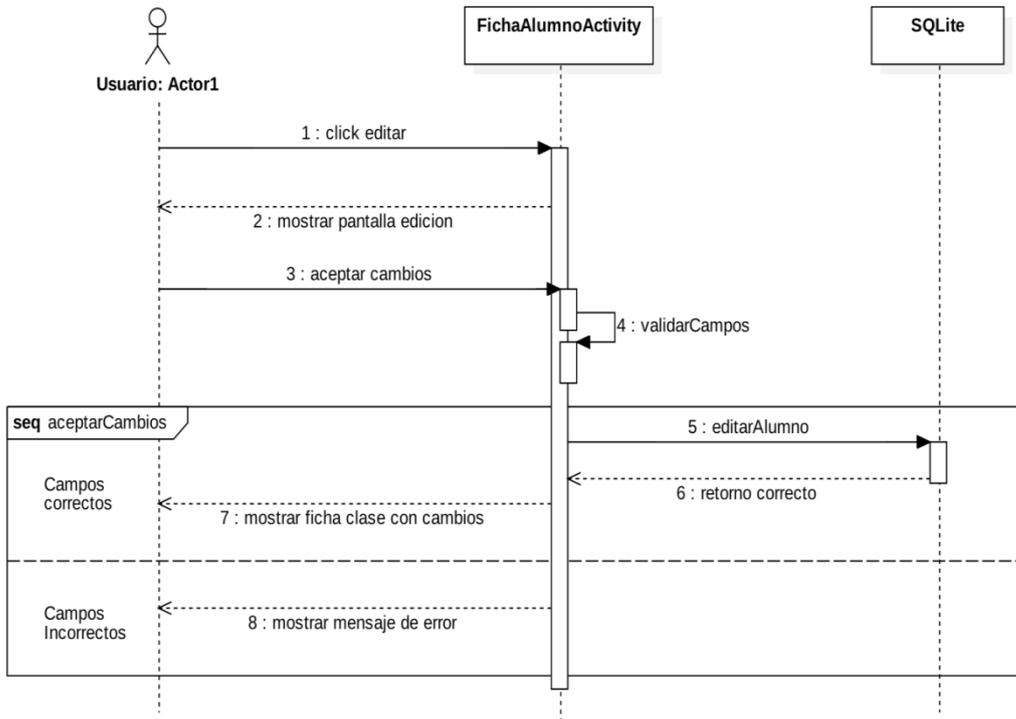


Figura 25.- Diagrama de secuencia - Eliminar objetivo.

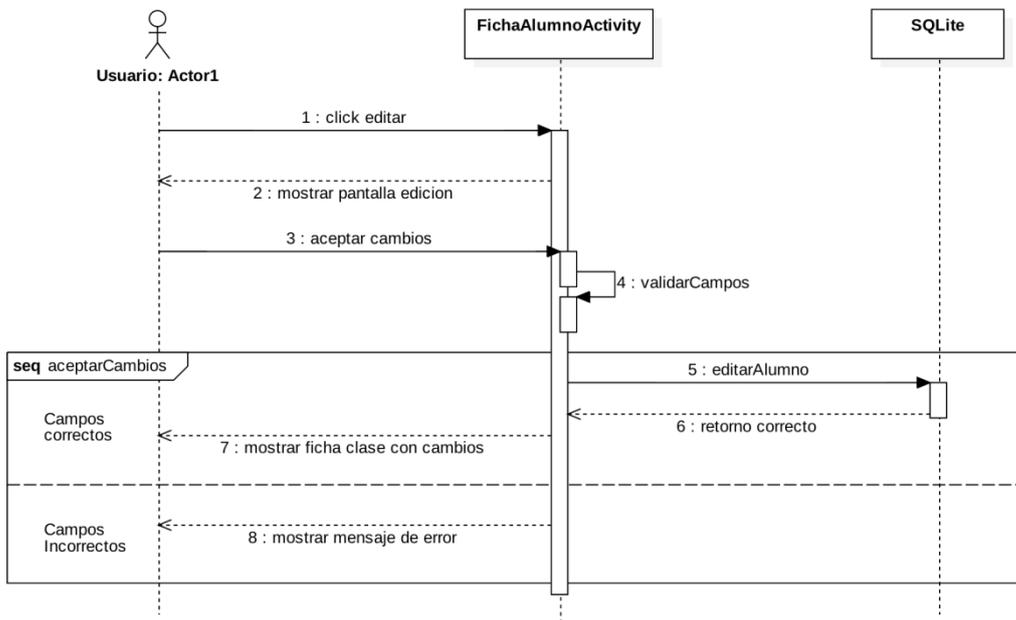


Figura 26.- Diagrama de secuencia - Editar alumno.

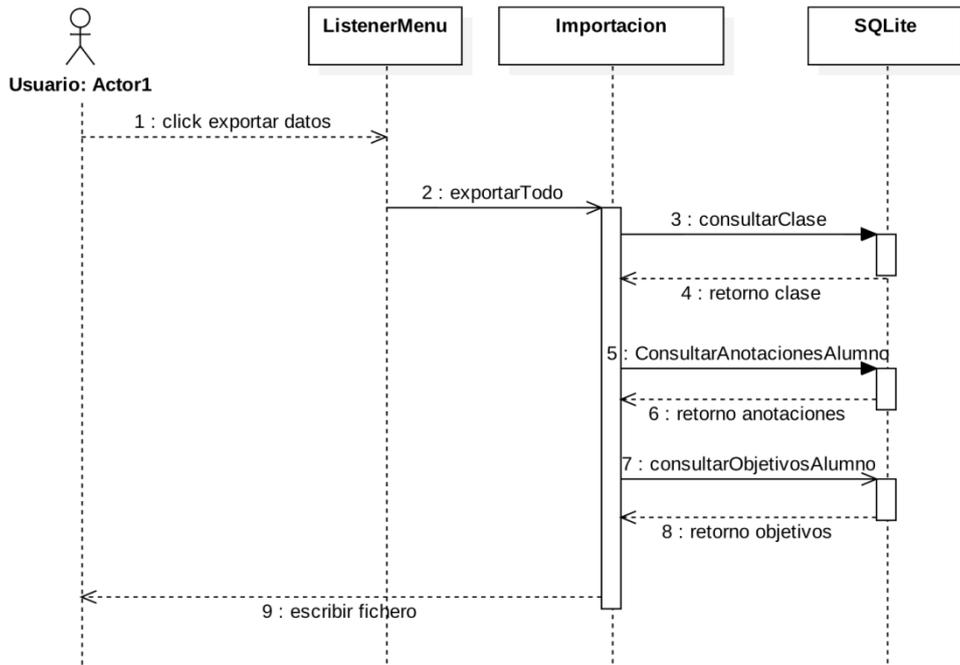


Figura 27.- Diagrama de secuencia - Exportar datos.

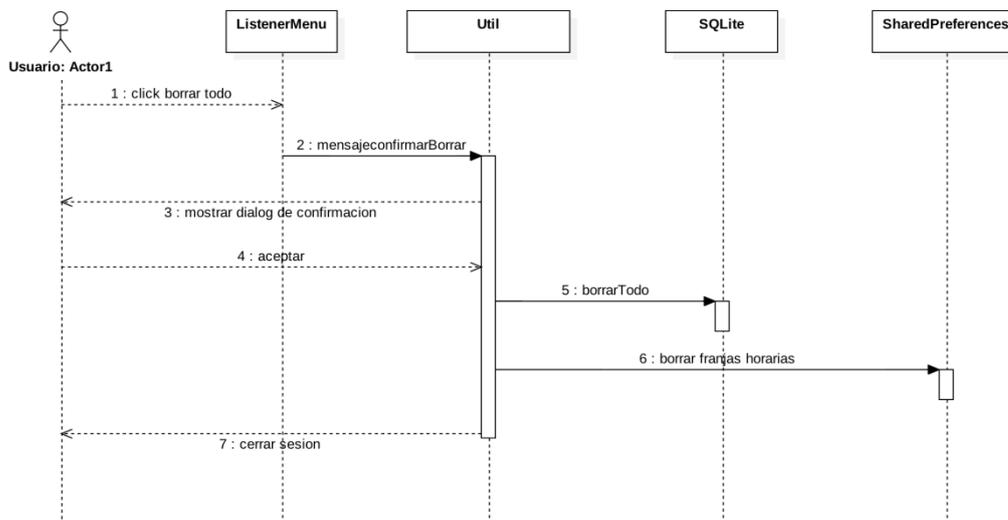


Figura 28.- Diagrama de secuencia - Eliminar todo.

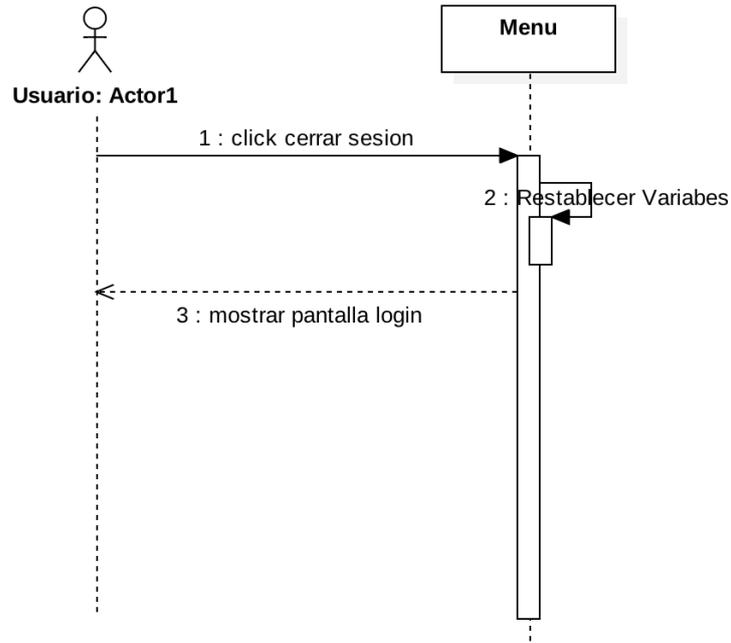


Figura 29.- Diagrama de secuencia - Cerrar sesión.

5.5. Modelo lógico

La estructura de la base de datos de la clase está definida en 6 tablas, detalladas a continuación:

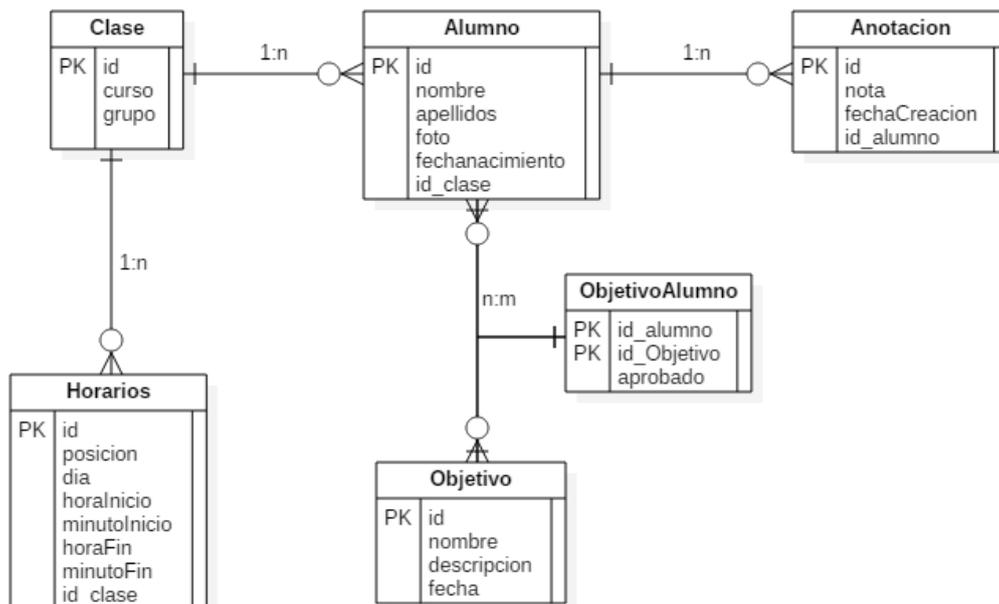


Figura 30.- Modelo Lógico, base de datos Clase.

La tabla Clase almacena todas las clases que imparte el docente. Esta tabla está relacionada con la tabla Horarios, que representa los espacios temporales que ocupa dicha clase.

A continuación se encuentra la tabla Alumnos, tabla principal del sistema, que almacena todos los alumnos a los que el docente imparte clase. Esta tabla está relacionada con la de Clases, en cuanto a que los alumnos se agrupan por clases y con la tabla anotación, en cuanto a que cada alumno tendrá asociado un grupo de anotaciones sobre su actividad.

La tabla Objetivos contiene todos los objetivos que el docente programa para sus clases y que permitirán al mismo realizar la evaluación del alumno junto con la tabla anotaciones, que almacena registros de la actividad del alumno, que complementan estos objetivos.

Por último la tabla objetivo alumno surge debido a la relación N:M entre las entidades alumno y objetivo. Esta tabla tendrá como PK la conjunción de las PK de alumno y objetivo, además del atributo propio aprobado.

La estructura de la base de datos de la alarma únicamente posee la tabla de alarmas, es la encargada del almacenaje de las alarmas del sistema definidas en el calendario:

Alarma	
PK	id
	titulo
	fecha
	hora
	repetir
	activa

Figura 31.- Modelo Lógico, base de datos Alarma.

Se permite el borrado de elementos de todas las tablas salvo de la tabla clase, ya que el borrado de una clase puede suponer pérdida de gran parte de información, puesto que al borrar la clase se borrarían todos sus elementos asociados (alumnos, objetivos, anotaciones y horarios). Con respecto al resto de tablas, si se elimina un elemento de clave principal, se localizan y eliminan los elementos asociados mediante clave foránea.

5.6. Diccionario de datos

En este apartado se presenta el diccionario de datos lógico basado en el diseño relacional. Se definirán los atributos que se encuentran en las tablas con una descripción de cada uno, que incluirá el tipo de datos y sus características.

El diccionario de datos es el siguiente:

- Tabla **alumno**

Campo	Tipo	Null	Clave	Otros
id	Integer(4)	NO	PK	Auto_Increment
nombre	Varchar(255)	NO		
apellidos	Varchar(255)	NO		
foto	Varchar(255)	NO		
fechaNacimiento	Varchar(255)	NO		
id_clase	Integer(4)	NO	FK	

Tabla 52.- Tabla alumno.

- Tabla **objetivo**

Campo	Tipo	Null	Clave	Otros
id	Integer(4)	NO	PK	Auto_Increment
nombre	Varchar(255)	NO		
descripcion	Varchar(255)	NO		
fecha	Varchar(255)	NO		

Tabla 53.- Tabla objetivo.

- Tabla **anotacion**

Campo	Tipo	Null	Clave	Otros
id	Integer(4)	NO	PK	Auto_Increment
nota	Varchar(255)	NO		
fechaCreacion	Varchar(255)	NO		
Id_alumno	Integer(4)	NO	FK	

Tabla 54.- Tabla anotacion.

- Tabla **clase**

Campo	Tipo	Null	Clave	Otros
id	Integer(4)	NO	PK	
curso	Integer(2)	NO		
grupo	Varchar(2)	NO		

Tabla 55.- Tabla clase.

- Tabla **horario**

Campo	Tipo	Null	Clave	Otros
id	Integer(4)	NO	PK	Auto_Increment
posicion	Integer(2)	NO		
dia	Varchar(20)	NO		
horaInicio	Integer(2)	NO		
minutoInicio	Integer(2)	NO		
horaFin	Integer(2)	NO		
minutoFin	Integer(2)	NO		
Id_Clase	Integer(4)	NO	FK	

Tabla 56.- Tabla horario.

- Tabla **objetivoalumno**

7	Tipo	Null	Clave	Otros
id_alumno	Integer(4)	NO	PK, FK	
id_objetivo	Integer(4)	NO	PK, FK	
aprobado	Integer(1)	NO		

Tabla 57.- Tabla objetivoalumno.

- Tabla **alarma**

Campo	Tipo	Null	Clave	Otros
id	Integer(4)	NO	PK	Auto_Increment
titulo	Varchar(255)	NO		
fecha	Varchar(255)	NO		
hora	Varchar(255)	NO		
repetir	Integer(1)	NO		
activa	Integer(1)	NO		

Tabla 58.- Tabla alarma.

5.7. Diseño de la interfaz

Este apartado proporciona una aproximación de lo que será la interfaz de usuario del sistema. Para la elaboración de este apartado se ha contado con Marlene Losada, ex-alumna del Grado de Magisterio en Educación Física.

Se han realizado diferentes entrevistas para determinar un primer diseño aproximado de la aplicación, tras lo cual se realizaron diversas pruebas con prototipos sobre papel con dos profesores del colegio Villalpando (Segovia). Estas sesiones fueron grabadas para su posterior análisis, tras el que se detectaron pequeñas dificultades con la interfaz de usuario, por lo que se decidió hacer un enlace directo desde la pantalla de horario para poder acceder a las clases.

En posteriores reuniones con los tutores del TFG, se decide, por temas de protección de datos, añadir un login que no estaba previsto en el diseño inicial de la aplicación, ya que ésta es de uso exclusivo y no admite múltiples usuarios desde un mismo terminal.

Finalmente en la exposición del TFG tras una reflexión de los docentes pertenecientes al tribunal se decide omitir la distinción entre objetivos específicos y generales, puesto que los objetivos generales serán cumplimentados mediante los específicos.

A continuación se mostrara el prototipo realizado a papel de la primera versión, el cual posee los cambios realizados por los tutores y los profesores del colegio Villalpando. Posteriormente se verá la versión final en la que se contempla también el cambio del tribunal.

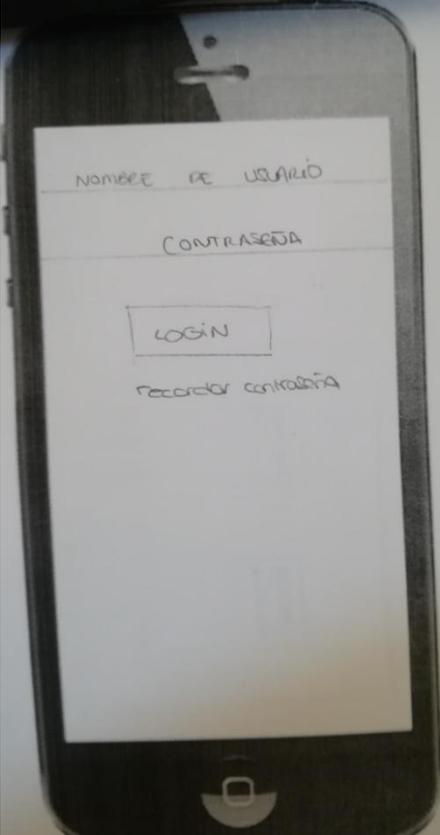
Nombre	Inicio de Sesión (Login)
Descripción	Página de inicio de la aplicación, contiene: <ul style="list-style-type: none"> • EditText para nombre de usuario • EditText para contraseña • Botón de Login • Enlace para recordar contraseña (o registro si es la primera apertura)
Activación	Se activa cuando el usuario abre la aplicación.
Boceto	
Eventos	<ul style="list-style-type: none"> • Al hacer login, entrar a la pantalla de inicio en caso de éxito. Mensaje de error en caso no encontrar el usuario o contraseña. • Envío de email para recordar contraseña o abrir pantalla de registro

Tabla 59.- Diseño pantalla "Inicio de Sesión (Login)".

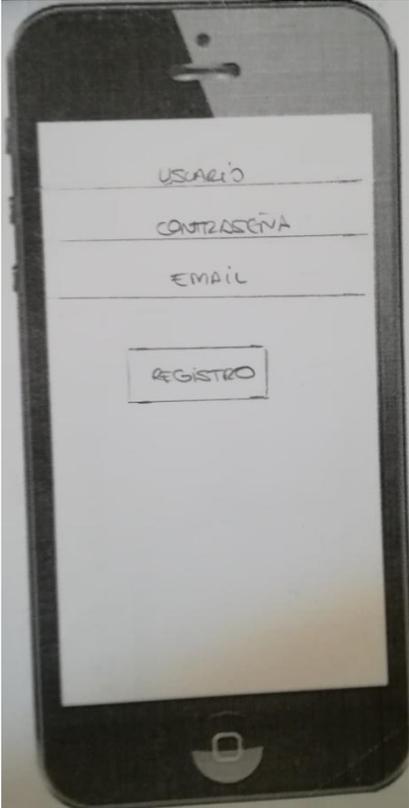
Nombre	Registro de usuario
Descripción	<p>Página de registro de la aplicación, contiene:</p> <ul style="list-style-type: none"> • EditText para nombre de usuario. • EditText para contraseña. • EditText para email. • Botón de Registro.
Activación	Se activa cuando el usuario selecciona la opción de registro.
Boceto	 <p>The sketch shows a mobile phone screen with three text input fields labeled 'USUARIO', 'CONTRASEÑA', and 'EMAIL' stacked vertically. Below these fields is a rectangular button labeled 'REGISTRO'.</p>
Eventos	<ul style="list-style-type: none"> • Al hacer clic en registrar, en caso de éxito abre la pantalla de login. En caso de error en los datos muestra mensaje.

Tabla 60.- Diseño pantalla "Registro de Usuario".

Nombre	Horario (Pantalla principal)
Descripción	Página de inicio de la aplicación cuando el usuario esta logueado, contiene: <ul style="list-style-type: none"> • Visualización del horario • Menú • Botón de calendario
Activación	Se activa cuando el usuario inicia sesión.
Boceto	
Eventos	<ul style="list-style-type: none"> • Acceso al menú. • Acceso al calendario. • Acceso a la lista de clases. • Acceso a la lista de alarmas.

Tabla 61.- Diseño pantalla "Horario (Pantalla principal)".

Nombre	Calendario
Descripción	<p>Pantalla que muestra el calendario con sus alarmas, contiene:</p> <ul style="list-style-type: none"> • Calendario. • Representación de alarmas mediante puntos.
Activación	Se activa cuando el usuario pulsa sobre el icono de calendario.
Boceto	
Eventos	<ul style="list-style-type: none"> • Volver al horario. • Crear alarma. • Acceso al menú. • Acceso a la lista de clases. • Acceso a la lista de alarmas.

Tabla 62.- Diseño pantalla "Calendario".

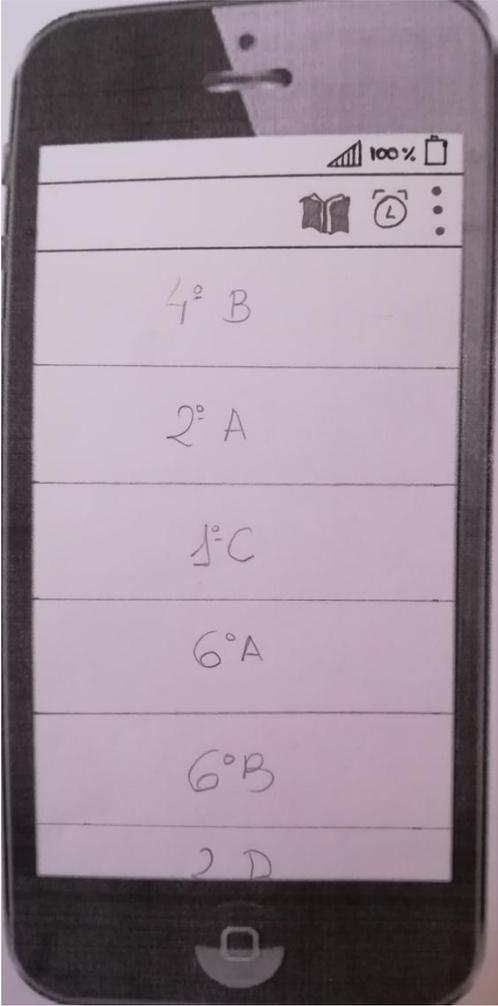
Nombre	Listado Clases
Descripción	<ul style="list-style-type: none"> • Pantalla que contiene un listado con las clases que el docente imparte, además del menú con sus respectivos botones.
Activación	Se activa desde el menú, icono libro.
Boceto	
Eventos	<ul style="list-style-type: none"> • Acceso a la pantalla de una clase, seleccionándola en la lista. • Acceso a la lista de las alarmas desde el menú. • Acceso a las acciones del menú.

Tabla 63.- Diseño pantalla "Listado Clases".

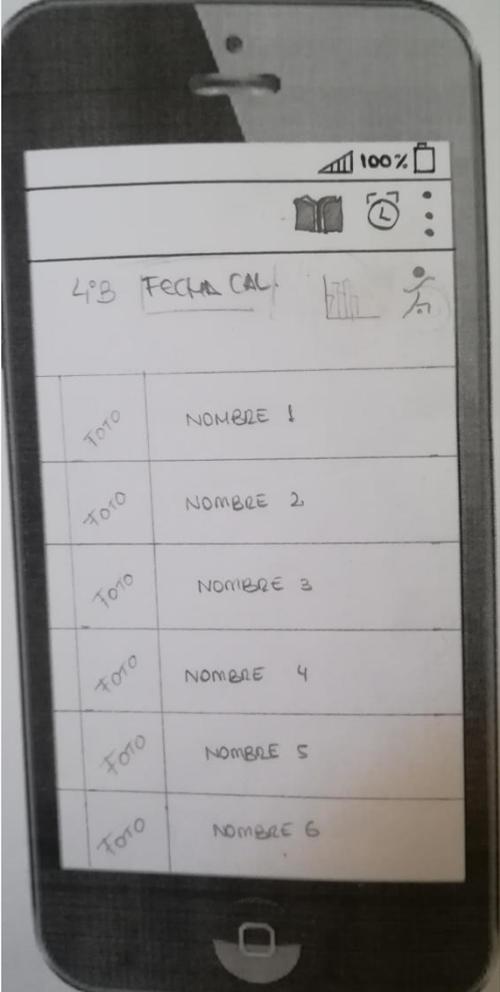
Nombre	Clase
Descripción	<ul style="list-style-type: none"> Pantalla que contiene la información de la clase, la lista de alumnos y un botón para asociar los objetivos.
Activación	Se activa cuando el usuario selecciona una clase, desde el listado de clases o desde el horario.
Boceto	
Eventos	<ul style="list-style-type: none"> Acceso a la pantalla del alumno haciendo seleccionando un alumno de la lista. Acceso al listado de objetivos para seleccionar un objetivo a la clase. Acceso a la lista de clases haciendo clic sobre el icono libro o volviendo hacia atrás. Acceso al listado de alarmas. Acceso a las opciones del menú.

Tabla 64.- Diseño pantalla "Clase".

Nombre	Ficha de Alumno
Descripción	<p>Pantalla de la ficha del alumno, además de los datos del alumno contiene:</p> <ul style="list-style-type: none"> • 2 Botones para crear anotación (tanto escrita como por voz) • Lista de objetivos asociados al alumno. • Lista de anotaciones asociadas al alumno. • ImageButton para seleccionar una foto del alumno.
Activación	Se activa cuando el usuario selecciona un alumno desde la pantalla de la clase.
Boceto	
Eventos	<ul style="list-style-type: none"> • Añadir anotación por voz. • Añadir anotación escrita. • Añadir foto al alumno. • Acceso a la lista de clases. • Acceso a la lista de alarmas. • Acceso a las opciones del menú.

Tabla 65.- Diseño pantalla "Ficha de Alumno".

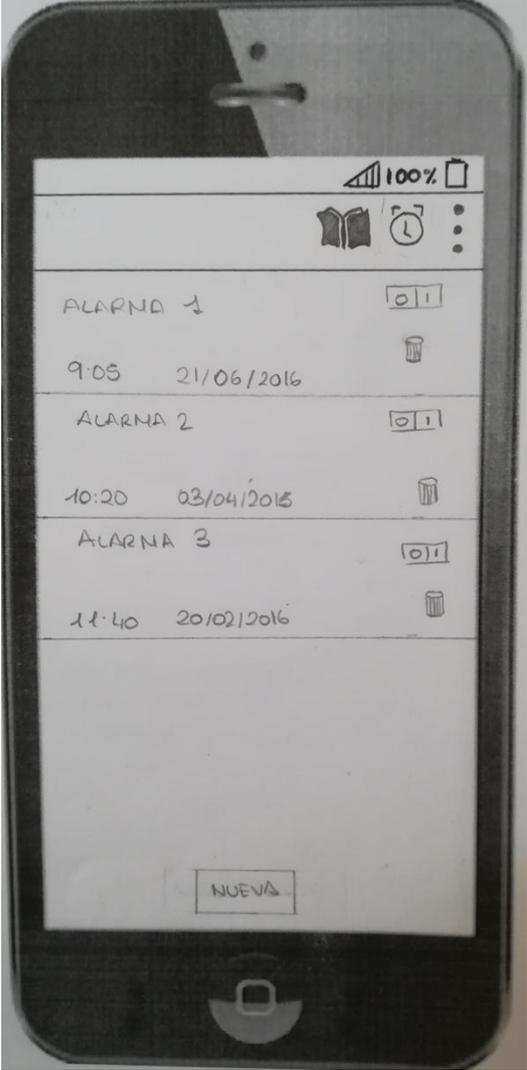
Nombre	Listado de Alarmas
Descripción	<ul style="list-style-type: none"> • Pantalla que contiene la lista de alarmas y un botón de activación y desactivación y otro para su eliminación.
Activación	Se activa cuando el usuario pulsa sobre el icono "despertador".
Boceto	
Eventos	<ul style="list-style-type: none"> • Desactivar/Activar alarma • Borrar alarma. • Crear nueva alarma. • Acceder al listado de clases. • Acceder a las opciones del menú.

Tabla 66.- Diseño pantalla "Listado de Alarmas".

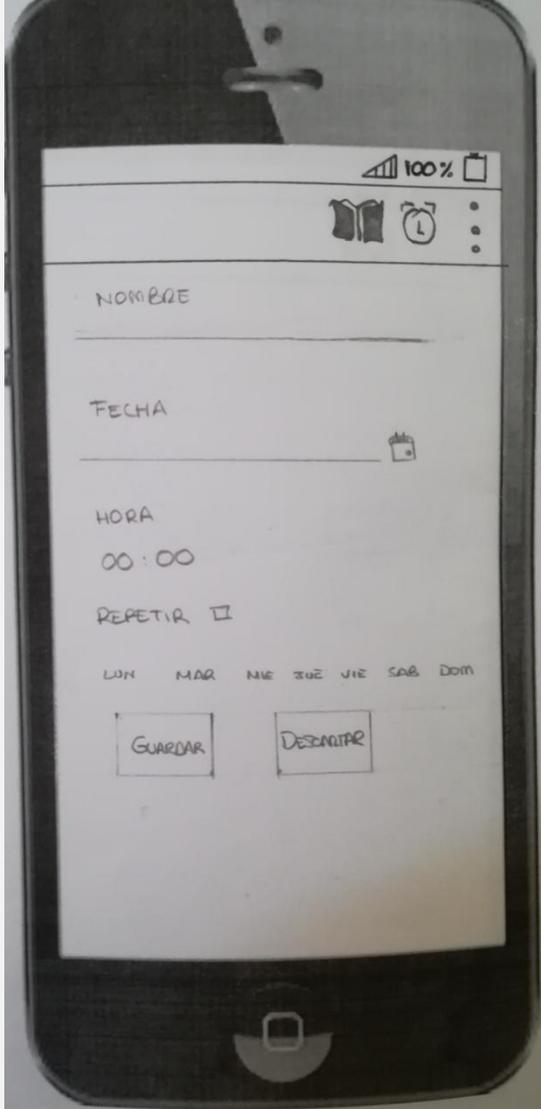
Nombre	Creación de Alarma
Descripción	Pantalla de creación de alarma, contiene: <ul style="list-style-type: none"> • EditText para nombre de la alarma. • EditText para seleccionar la fecha de la alarma. • EditText para seleccionar la hora de la alarma. • Check box para seleccionar su repetición. • Botones de guardar y cancelar
Activación	Se activa al pulsar sobre la opción de menú nueva alarma.
Boceto	
Eventos	<ul style="list-style-type: none"> • Guardar alarma si tiene éxito se guardara, si no mostrara un mensaje con el motivo. • Descartar alarma retornara al activity anterior. • Acceso a las opciones del menú. • Acceso al listado de clases-

Tabla 67.- Diseño pantalla "Creación de Alarma".

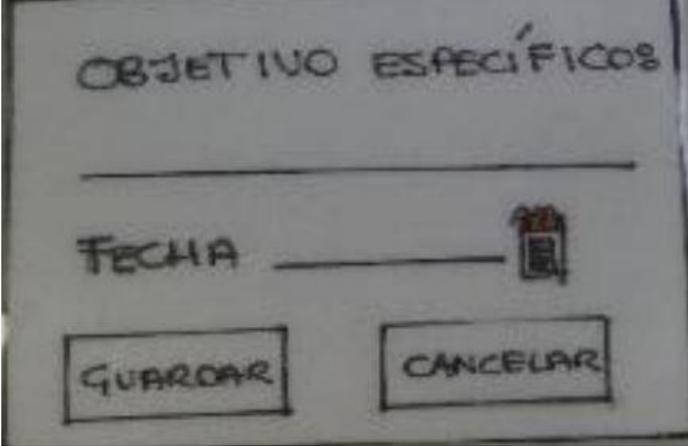
Nombre	Creación de Objetivo
Descripción	<p>Pantalla de creación del objetivo, contiene:</p> <ul style="list-style-type: none"> • EditText para nombre del objetivo. • EditText para seleccionar la fecha del objetivo. • Botones de guardar y cancelar
Activación	Se activa al pulsar sobre la opción de menú nuevo objetivo.
Boceto	
Eventos	<ul style="list-style-type: none"> • Guardar objetivo si tiene éxito se guardará, sí no mostrara un mensaje con el motivo.

Tabla 68.- Diseño pantalla "Creación de Objetivo".

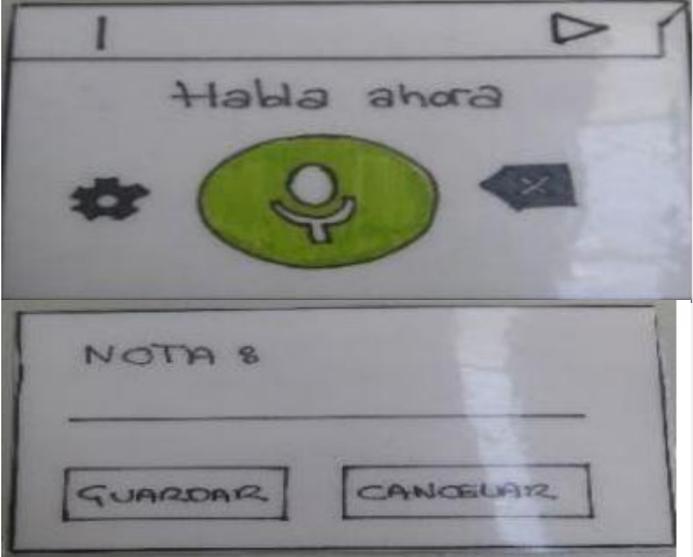
Nombre	Creación de Anotaciones
Descripción	Dialogs de creación de anotaciones.
Activación	Se activa al pulsar en la ficha del alumno la opción de grabar o añadir nota.
Boceto	
Eventos	<ul style="list-style-type: none"> • Guardar anotación si tiene éxito se guardará, sí no mostrara un mensaje con el motivo.

Tabla 69.- Diseño pantalla "Creación de Anotaciones".

A continuación se muestra las pantallas finales resultantes de estos bocetos.

En EFAIDia solo podrá haber un usuario por terminal, es necesario un registro para poder acceder a cualquier funcionalidad de la aplicación.

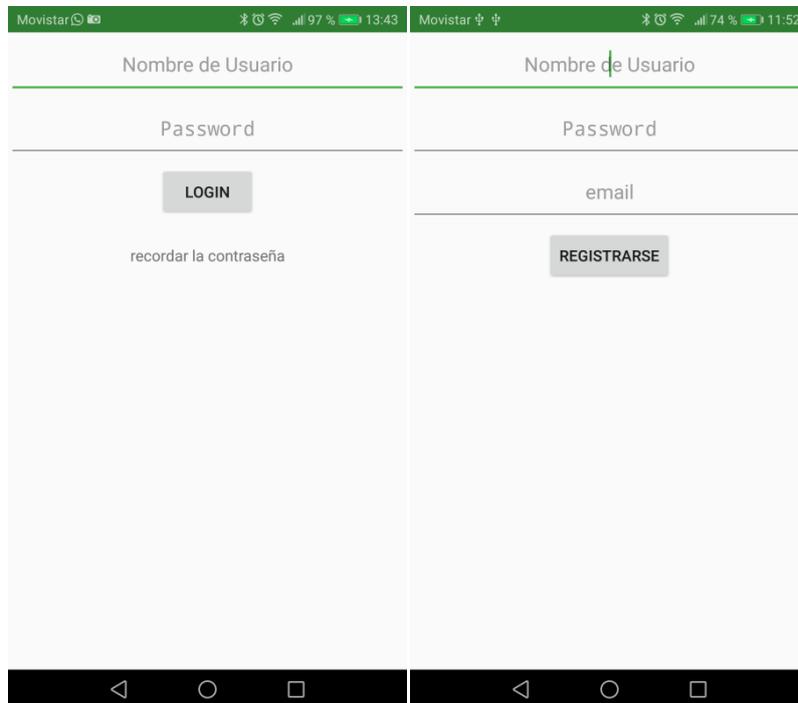


Figura 32.- Registro y Login.

En el primer registro se solicita al usuario que complete un formulario, en el cual debe indicar las horas que conformarán su horario de clase, incluyendo recreos.

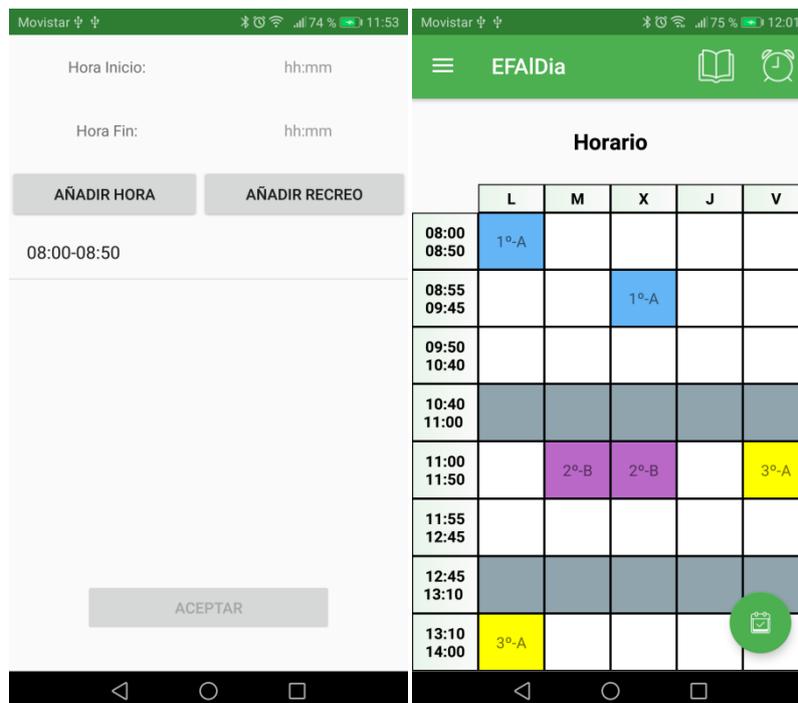


Figura 33.- Configuración de horario y Horario.

En el calendario se verán representadas con un punto verde las alarmas creadas. En el menú veremos todas las funcionalidades disponibles por la aplicación.

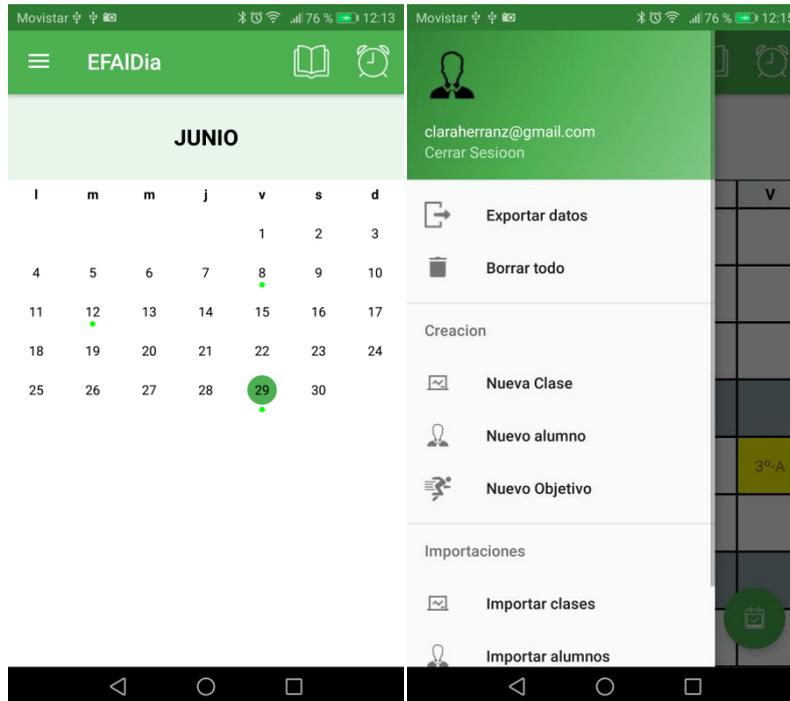


Figura 34.- Calendario y Menú.

En el primer activity vemos los campos de creación de la alarma y en el segundo activity podemos ver como se representa el listado de alarmas, dónde podrán ser borradas y activadas/desactivadas.

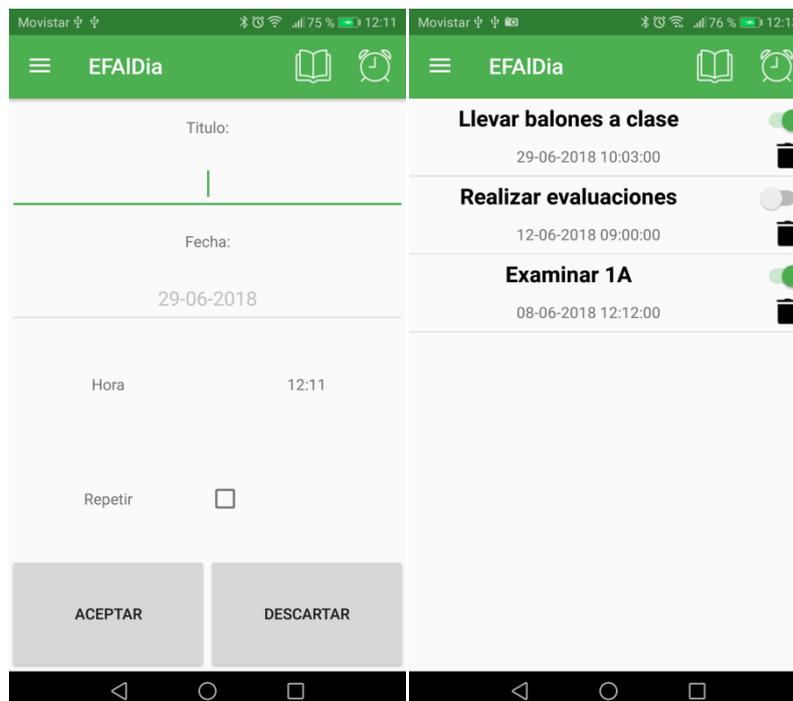


Figura 35.- Creación de alarmas y Listado de Alarmas.

Estas dos pantallas representan la creación de la clase y el listado de todas las clases existentes, al que se puede acceder desde el icono “libro”.

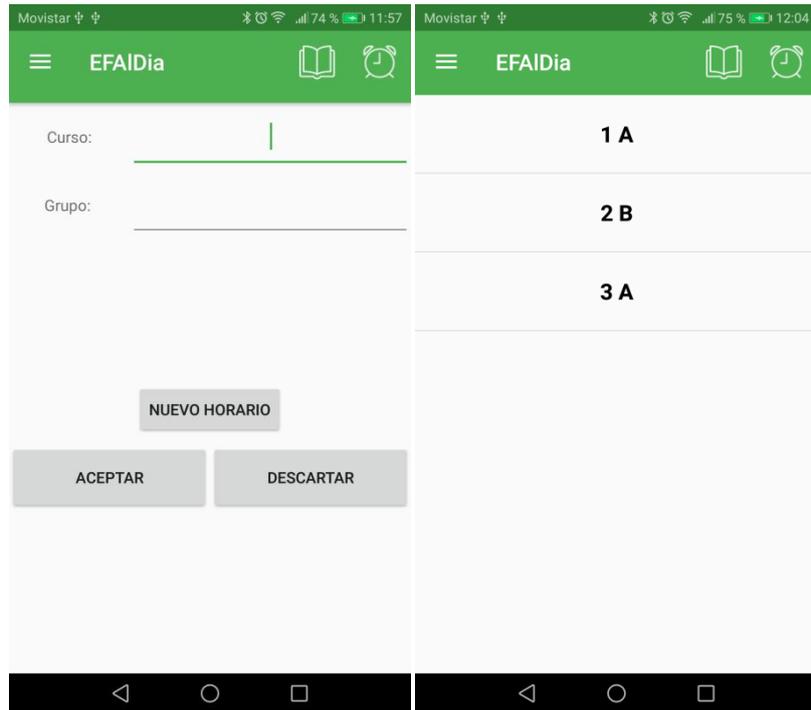


Figura 36.- Crear Clase y Listado de Clases.

En el primer activity vemos la creación de un alumno, en el segundo podemos observar la representación de una clase, la cual tiene un listado de los alumnos que la forman.

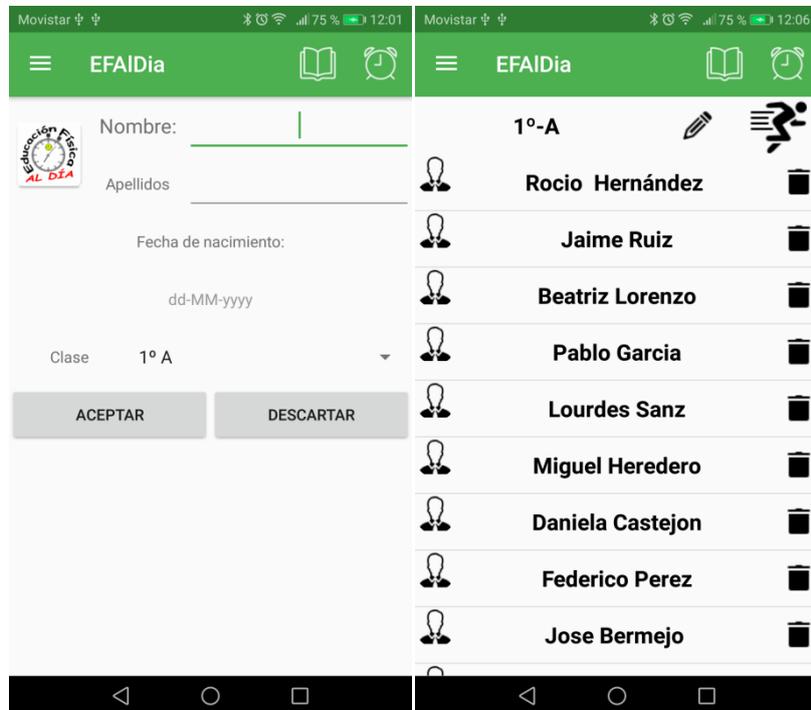


Figura 37.- Crear Alumno y Clase.

El objetivo, en su creación, no está asociado a ninguna clase ni a ningún alumno, esto se realizará a través del dialog de asociación del objetivo, bien lanzado desde una clase, lo que hará que todos los alumnos que componen la clase tengan el objetivo, o lanzándolo desde un alumno concreto, asociando el objetivo únicamente a dicho alumno.

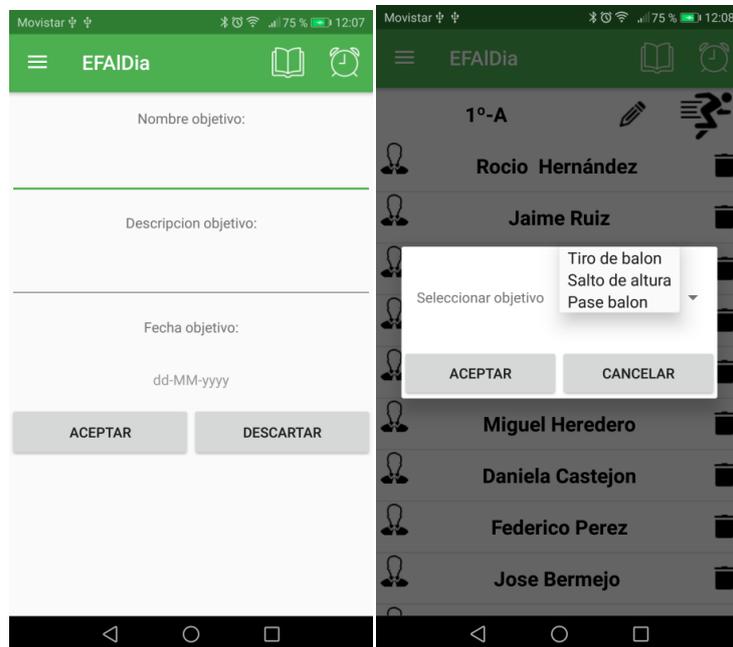


Figura 38.- Creación de Objetivo y su asociación.

Este es el activity principal de la aplicación, en este activity se verán los objetivos y las anotaciones del alumno, lo que facilitará la evaluación.



Figura 39.- Ficha Alumno.

Por último, vemos los dialogs utilizados para la creación de anotaciones, éstas pueden realizarse de dos maneras, la primera escribiendo directamente la observación deseada del alumno y la segunda haciendo una grabación de voz, que se transcribirá y se aprobará para su guardado.

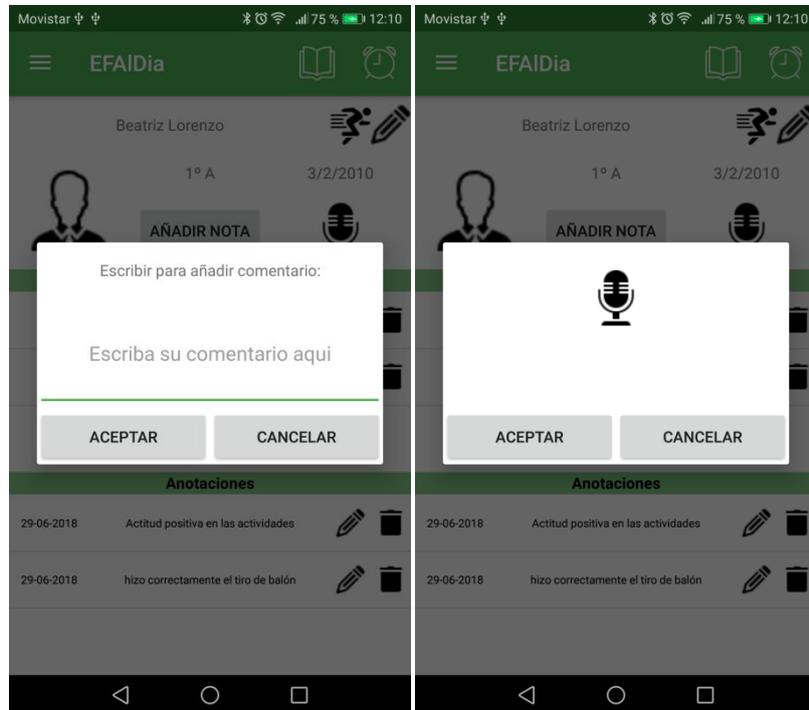


Figura 40.- Creación de anotaciones.

Capítulo 6 - Implementación.

En este capítulo se explica cómo se llevó a cabo la implementación de la aplicación móvil. Primero describiremos las tecnologías utilizadas para su funcionamiento y posteriormente veremos las herramientas usadas.

6.1. Tecnologías utilizadas

Esta aplicación está destinada para dispositivos móviles con sistema operativo Android, por lo cual está desarrollada en Android Studio. Su desarrollo se realiza en lenguaje Java, con pequeñas variaciones que presenta como los activities, layouts, servicios etc.

El diseño de los activities se realiza en XML, aunque Android Studio dispone de un simulador gráfico para el diseño, en el cual puedes incorporar directamente los componentes de los activities. Para el diseño de los XML se ha usado LinearLayout, que alinea los campos en una dirección única (horizontal o vertical). A continuación se definen los campos más utilizados para formar los XML:

- `Layout_width`, `layout_height`, `layout_weight` y `weightSum`: definen el ancho y el largo y el peso de uno de los anteriores en caso de división (`layout_weight` en caso de un componente y `weightSum` para definir el total de un layout).
- `Orientation`: indica la orientación que seguirá el `LinearLayout` (horizontal o vertical).

Para ver su uso se muestra el XML correspondiente al activity de registro, el primero usado al instalar la aplicación:

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
android:weightSum="100"

tools:context="com.example.clara.efaldia.activity.RegisterActivity">
    <EditText
        android:layout_width="match_parent"
android:layout_height="0dip"
        android:layout_weight="10"
        android:inputType="textPersonName"
        android:layout_below="@+id/edName"
        android:gravity="center|center_horizontal"
        android:id="@+id/edUserName"
        android:hint="@string/nombre_de_usuario"
        tools:ignore="HardcodedText,UnknownId" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="0dip"
        android:layout_weight="10"
        android:layout_below="@+id/edUserName"
        android:gravity="center|center_horizontal"
        android:id="@+id/edPassword"
        android:inputType="textPassword"
        android:hint="@string/password" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="0dip"
        android:layout_weight="10"
        android:gravity="center|center_horizontal"
        android:inputType="textEmailAddress"
        android:id="@+id/edEmail"
        android:hint="email"
        tools:ignore="HardcodedText"
        android:layout_below="@+id/edPassword"/>

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="0dip"
        android:layout_weight="10"
        android:orientation="horizontal"
        android:gravity="center|center_horizontal">
        <Button
            android:text="@string/registrarse"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:id="@+id/btRegister"
            android:onClick="registrar"
            android:layout_below="@+id/edEmail"
            android:layout_alignParentLeft="true"
            android:layout_alignParentStart="true"
            android:layout_alignParentRight="true"
            android:layout_alignParentEnd="true" />
    </LinearLayout>

</LinearLayout>

```

Para la transcripción de voz se ha utilizado la API de Google Speech-To-Text, que permite la conversión de audio a texto mediante modelos de redes neuronales. Es una API fácil de usar y reconoce 120 idiomas.

En el siguiente código se observa la creación del método hablar, que lanza el dialog encargado de realizar la grabación. A través de ACTION_RECOGNIZE_SPEECH lanza el activity que solicita al usuario hablar y lo envía para ser procesado por el speech recognizer. Posteriormente, se le informa mediante EXTRA_LANGUAGE_MODEL qué idioma se va a usar para la transcripción, este paso es obligatorio. Finalmente se captura el resultado de la actividad, y, en caso de excepción, se informa al usuario de que no dispone del reconocimiento de voz.

```
//metodo que se lanza cuando abro el dialog de grabación de voz
public void Hablar(View v) {

    Intent intentActionRecognizeSpeech = new
    Intent (RecognizerIntent.ACTION_RECOGNIZE_SPEECH);

    //Configura el Lenguaje (Español)
    intentActionRecognizeSpeech.putExtra (RecognizerIntent.EXTRA_LANGUAGE_MODEL, "es-ES");
    try {
        startActivityForResult (intentActionRecognizeSpeech,RECOGNIZE_SPEECH_ACTIVITY);
    } catch (ActivityNotFoundException e) {
        Toast.makeText (getApplicationContext (),
        "No dispones de reconocimiento de voz", Toast.LENGTH_SHORT).show ();
    }
}
```

Para los eventos de las alarmas se han usado las notificaciones de Android, ya que Android permite poner notificaciones en la barra de titulo de la aplicación. En concreto se ha usado NotificationManager, que permite recibir el contexto de la alarma a través del método getSystemService().

La siguiente secuencia de código muestra el método onRecibe, que contiene el Bundle para enviar los datos de la alarma, NotificationCompat para construir los diseños de notificación, dar el formato y poner el texto en el título, contenido e imagen. Por último, NotificationManager, encargado de lanzar el evento con el método notify.

```
//Metodo para lanzar el evento de la alarma
public void onReceive(Context context, Intent intent) {
    Bundle bundle = intent.getBundleExtra("bundle");
    Alarma a = (Alarma) bundle.getSerializable("alarma");
    // se usa NotificationCompat para ganar compatibilidad con versiones mas
    antiguas sin metodos obsoletos
    NotificationCompat.Builder mBuilder =
        new NotificationCompat.Builder(context);

    NotificationCompat.BigTextStyle bigText = new NotificationCompat.BigTextStyle();
    //En el evento que lanzamos imprimimos el titulo de la alarma como contenido
    bigText.bigText(a.getTitulo());
    //Ponemos de titulo el texto alarma junto con la hora
    bigText.setBigContentTitle("Alarma" + a.getStrHora());
    bigText.setSummaryText(a.getTitulo());
    //ponemos el icono de la aplicacion en la notificacion
    mBuilder.setSmallIcon(R.drawable.reloj2);
    mBuilder.setContentTitle("Your Title");
    mBuilder.setContentText("Your text");
    mBuilder.setPriority(Notification.PRIORITY_MAX);
    mBuilder.setStyle(bigText);
    //se crea la instancia NotificationManager se encarga de gestionar las
    notificaciones
    NotificationManager mNotificationManager =
        (NotificationManager)
    context.getSystemService(Context.NOTIFICATION_SERVICE);
    // se publica en la barra de estado
    mNotificationManager.notify(0, mBuilder.build());
}
}
```

Para el calendario se ha utilizado CompactCalendarView de SundeepK⁽¹⁾. Es una vista de calendario simple, está basado en las clases de fecha y calendario de Java, lo que proporciona un API sencillo para la consulta de fechas y creación de alarmas.

6.2. Herramientas utilizadas

En este apartado se definen las herramientas que han sido utilizadas en el desarrollo del sistema:

- **Android OS:** sistema operativo basado en Linux. Fue diseñado principalmente para dispositivos móviles con pantalla táctil (Smartphones, tablets o relojes). Es un sistema operativo muy intuitivo, lo que le ha llevado a tener la mayor cuota de mercado.
- **Android Studio (versión 3.0.1):** entorno de desarrollo integrado oficial para la plataforma Android. Proporciona las herramientas para crear aplicaciones en cada tipo de dispositivo Android. Posee soporte para desarrolladores y gran cantidad de información (guías y documentación).
- **SQLite:** como ya se ha definido anteriormente, es un sistema de gestión de bases de datos relacionales compatible con ACID.
- **Google Speech-To-Text API:** API que permite a los desarrolladores convertir audio en texto. Puede transcribir en tiempo real o datos pregrabados, utilizando la tecnología de aprendizaje de Google.
- **SQLiteStudio (versión 3.1.0):** interfaz gráfica para bases de datos SQLite, potente, ligero, rápido e intuitivo. No requiere instalación, es de código abierto, permite exportar a diferentes formatos e importar desde CSV.

(1) Disponible en: <https://github.com/SundeepK/CompactCalendarView>

6.3. Estructura del proyecto

En este apartado se explicará cómo se estructura el proyecto. Los ficheros de la aplicación se encuentran en la ruta “AndroidStudioProjects\EFALDia\app\src\main”

Dentro de esta ruta se encuentran dos directorios, Java y Res. Dentro del directorio Java encontraremos las clases java de la aplicación, divididas en distintos directorios, en los que se han agrupado las diferentes clases:

- **java\com\example\clara\efaldia\activity:** aquí encontraremos los ficheros .java asociados a las activities.
- **java\com\example\clara\efaldia\listener:** este directorio contiene el .java con el listener del menú deslizable .
- **java\com\example\clara\efaldia\modelo:** en este fichero se encuentran las clases pertenecientes a las entidades básicas.
- **java\com\example\clara\efaldia\sqlite:** se encarga de lo relativo a la base de datos, encontraremos su estructura y métodos.
- **java\com\example\clara\efaldia\util:** contiene funcionalidades usadas desde las clases de los activity.

Dentro del directorio Res, se encuentra todo lo relativo a los recursos. Sus directorios más importantes son:

- **drawable:** contiene todos los recursos visuales de la aplicación (las imágenes usadas en botones, el logo etc.)
- **layout:** contiene los layouts correspondientes a los .java del directorio activity. Por cada java hay dos layouts (uno horizontal y uno vertical).
- **menu:** contiene los menús de la aplicación.

Capítulo 7 - Pruebas

En este capítulo hablaremos de las pruebas realizadas durante y después del desarrollo de la aplicación móvil. Se han realizado distintas pruebas de caja negra a lo largo de todo el proceso.

7.1. Pruebas de caja negra

Están destinadas a la comprobación el correcto funcionamiento de las funcionalidades del sistema, para que realicen las acciones que se han definido y devuelvan los resultados esperados.

Por lo tanto en este punto se analizarán funcionalidades completas de la aplicación móvil para comprobar que satisfacen los requisitos establecidos en el análisis. Algunas de las pruebas realizadas durante la implementación han sido:

Identificador	Identificación de usuario (login)
Objetivo	Identificar al usuario en la aplicación.
Prerrequisito	El usuario debe haberse registrado previamente en la aplicación.
Datos de entrada	Campos pedidos en el formulario (nombre y contraseña).
Resultado esperado	Usuario identificado en el sistema con acceso a las funcionalidades de este.
Resultado obtenido	Correcto.

Tabla 70.- PCN Identificación de usuario.

Identificador	Cerrar sesión
Objetivo	Cerrar sesión del usuario en la aplicación.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación.
Datos de entrada	Ninguno (Ya que solo hay un usuario por terminal)
Resultado esperado	Usuario vuelve a la pantalla de Login y no tiene acceso a ninguna funcionalidad del sistema.
Resultado obtenido	Correcto.

Tabla 71.- PCN Cerrar sesión.

Identificador	Creación de alarma
Objetivo	Crear alarma en la aplicación.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación.
Datos de entrada	Campos pedidos en el formulario (titulo, hora, repetir)
Resultado esperado	Alarma creada en el sistema y visualizada en el calendario.
Resultado obtenido	Error, no se visualiza la alarma en el calendario.
Cambios realizados	Dar formato a la fecha para que coincida con el calendario
Resultado esperado	Alarma creada en el sistema y visualizada en el calendario.
Resultado obtenido	Correcto.

Tabla 72.- PCN Creación de alarma.

Identificador	Borrar todo
Objetivo	Borrar todos los datos de la base de datos y de las preferencias compartidas salvo los datos del usuario.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe seleccionar del menú la opción "borrar todo".
Datos de entrada	
Resultado esperado	Datos borrados y cierre de sesión.
Resultado obtenido	Correcto.

Tabla 73.- PCN Borrar todo.

Identificador	Visualizar ficha alumno.
Objetivo	Ver la ficha del alumno, con sus respectivos mensajes y objetivos.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. Selección de un alumno.
Datos de entrada	Id del alumno seleccionado.
Resultado esperado	Mostrar ficha del alumno. Cargar sus objetivos. Cargar sus anotaciones
Resultado obtenido	Correcto.

Tabla 74.- PCN Visualizar ficha alumno.

Identificador	Asociación de objetivo.
Objetivo	Asociar un objetivo a una clase.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe seleccionar desde el la ficha de la clase el icono “corredor” y escoger el objetivo deseado.
Datos de entrada	Id objetivo seleccionado.
Resultado esperado	Objetivo asociado a todos los alumnos de la clase.
Resultado obtenido	Correcto.

Tabla 75.- PCN Asociación de objetivo.

Identificador	Visualizar alumnos.
Objetivo	Visualizar la lista de los alumnos de una clase.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe haber seleccionado una clase.
Datos de entrada	Id claseseleccionado.
Resultado esperado	Visualización de la lista de alumnos.
Resultado obtenido	Correcto.

Tabla 76.- PCN Visualizar alumnos.

Identificador	Registrar usuario.
Objetivo	Realizar el registro del usuario en la aplicación.
Prerrequisito	Debe ser la primera apertura de la aplicación.
Datos de entrada	Campos del formulario de registro.
Resultado esperado	Usuario creado en la aplicación.
Resultado obtenido	Correcto.

Tabla 77.- PCN Registrar usuario.

Identificador	Importar clases.
Objetivo	Importar clases desde .
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe haber creado el fichero de importación de las clases y haberlo situado en la carpeta de importaciones.
Datos de entrada	Campos del fichero.
Resultado esperado	Clases creadas en el sistema.
Resultado obtenido	Correcto.

Tabla 78.-PCN Importar clases.

Identificador	Editar foto del alumno.
Objetivo	Realizar foto y cambiar la existente en el alumno.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe encontrarse en la ficha del alumno. El usuario debe seleccionar la foto o icono del alumno.
Datos de entrada	Id del alumno. Ruta de la foto realizada.
Resultado esperado	Foto cambiada en el perfil del alumno y almacenada en la base de datos.
Resultado obtenido	Error. La foto está guardada en la base de datos pero no se visualiza tras su inserción.
Cambios realizados	Se incluye un for result para que si realiza esta acción se refresque la ficha del alumno y se visualice la foto.
Resultado esperado	Foto guardada y visualizada en el alumno.
Resultado obtenido	Correcto

Tabla 79.- PCN Editar foto del alumno.

Identificador	Exportar datos.
Objetivo	Realizar foto y cambiar la existente en el alumno.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe seleccionar la opción de menú "Exportar datos".
Datos de entrada	
Resultado esperado	Creación de fichero de texto en la carpeta de exportación con los datos de las clases y los alumnos.
Resultado obtenido	Correcto.

Tabla 80.- PCN Exportar datos.

Identificador	Lanzar evento de alarma.
Objetivo	Al cumplirse la fecha de la alarma lanzar un evento.
Prerrequisito	El usuario debe haber creado una alarma con anterioridad. La alarma debe estar activa.
Datos de entrada	Id alarma.
Resultado esperado	Al llegar a la fecha indicada en la alarma lanzarse un evento con el contenido de la alarma y el logo de la aplicación.
Resultado obtenido	Error. El evento se lanza pero los datos no se visualizan correctamente.
Cambios realizados	Se incluyen nuevas etiquetas en las que se indican los campos capturados de la alarma.
Resultado esperado	Al llegar a la fecha indicada en la alarma lanzarse un evento con el contenido de la alarma y el logo de la aplicación.
Resultado obtenido	Correcto

Tabla 81.- Lanzar evento alarma.

Identificador	Desactivar alarma.
Objetivo	Desactivación de una alarma.
Prerrequisito	El usuario debe haberse registrado e identificado previamente en la aplicación. El usuario debe encontrarse haber creado una alarma. El usuario debe seleccionar la desactivación de la alarma desde su listado.
Datos de entrada	Id alarma.
Resultado esperado	Alarma desactivada, ya no se visualiza en el sistema ni se generan sus eventos.
Resultado obtenido	Error. No se modifica su estado en la base de datos.
Cambios realizados	Se modifica el método de la base de datos cambiarEstadoAlarmas tras ver mediante la depuración que no realiza su funcionalidad.
Resultado esperado	Alarma desactivada, ya no se visualiza en el sistema ni se generan sus eventos.
Resultado obtenido	Error. Se modifica su estado en la base de datos pero se sigue visualizando en el calendario.
Cambios realizados	Se modifica el método que carga el calendario para que no pinte las alarmas con estado desactivado.
Resultado esperado	Alarma desactivada, ya no se visualiza en el sistema ni se generan sus eventos.
Resultado obtenido	Correcto.

Tabla 82.- PCN Desactivar alarma.

SECCIÓN 3 – MANUALES

Capítulo 8 - Manuales

En este capítulo se encuentran los manuales, tanto el de instalación como el de uso de la aplicación.

8.1. Manual de instalación

El método habitual para instalar la este tipo de aplicaciones, es su descarga desde el Play Store de Google, su instalación suele ser rápida y sencilla. Para esta aplicación como no se ha creado una cuenta en el Google Play Store, se deberá instalar desde el APK que contiene el CD adjuntado a la memoria.

8.2. Manual de usuario.

Toda aplicación requiere de un manual de usuario que facilite el aprendizaje de la misma. También servirá de punto de referencia en caso de que el usuario no sepa utilizar alguna de sus funcionalidades.

- **Registro:** es la primera pantalla que aparecerá al descargar la aplicación, sin registro no se podrá acceder a ninguna de las funcionalidades de la aplicación móvil. Para registrarnos correctamente deberemos rellenar todos los campos que pide la pantalla (nombre de usuario, contraseña y email). Se comprobará que el email sea correcto.

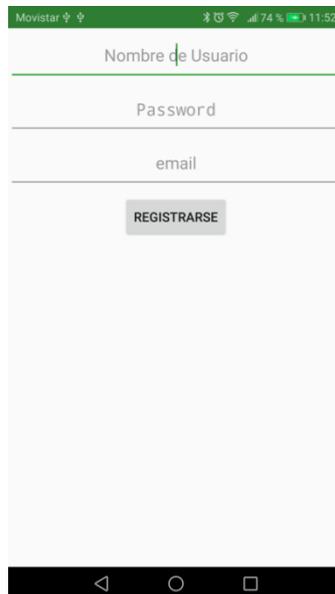


Figura 41.- Captura de registro.

- **Login:** una vez el usuario se haya registrado en la aplicación, se abrirá la pantalla de login. En esta pantalla únicamente será necesario introducir el nombre del usuario y la contraseña. También dispone de un enlace para solicitar el recordatorio de la contraseña, en caso de olvido. Al pulsar en este enlace se enviará un email al correo indicado en el registro, informando del usuario y la contraseña.

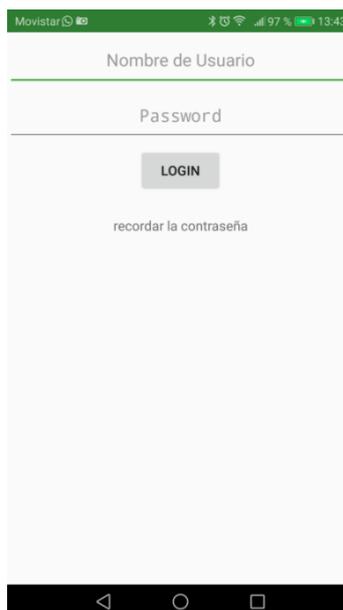


Figura 42.- Captura de Login.

- **Configuración horario:** tras el primer login en la aplicación, se abrirá una pantalla para configurar el horario que aparecerá en la pantalla principal de la aplicación. En esta pantalla se crearán las franjas horarias que conformarán las clases y los recreos. Se deberá indicar una hora de inicio y una hora de fin y posteriormente se deberá indicar si esa franja pertenece a un recreo o a un horario de clase. Una vez completada esta configuración se activará el botón para aceptar la definición del horario.

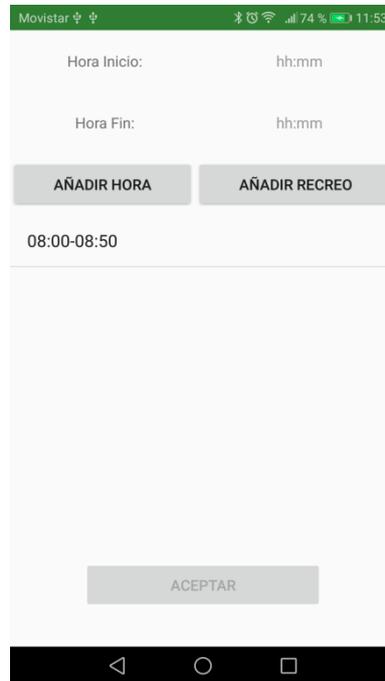


Figura 43.- Captura configuración horario.

- **Pantalla principal:** tras la configuración, iremos a la pantalla del horario, en la que veremos el horario que se ha definido anteriormente, dibujado sin clases. Para comenzar la primera carga de datos iremos al menú lateral (disponible en todas las pantallas) desplazando el dedo hacia la derecha de la pantalla. Se ofrecerán dos opciones para la creación de alumnos, clases y objetivos. La primera será crearlas manualmente desde la aplicación y la segunda será mediante la importación de ficheros de texto.

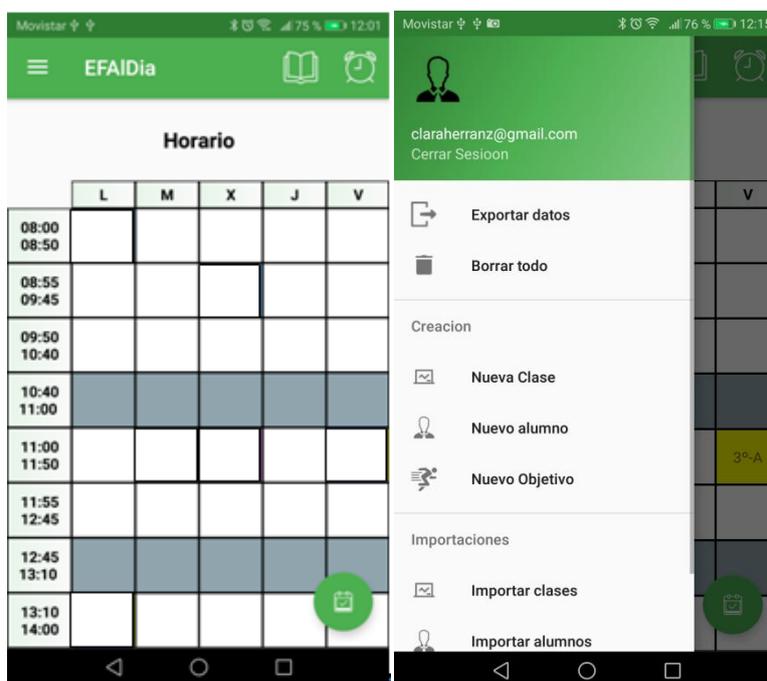


Figura 44.- Captura de Pantalla principal y Menú.

- **Carga inicial por importación de datos:** si seleccionamos la opción de importación, los ficheros deberán contener las siguientes características:
 - Los ficheros sólo deberán contener datos, en ningún momento deberán incluir cabeceras.
 - Deberán almacenarse en la carpeta de la aplicación denominada "Importar".
 - **Fichero importación clases:** Las clases no tendrán un id autoincremental, con el objetivo de poder asociar en la importación de los alumnos la clase a la que pertenecen. El fichero deberá llamarse clases y sus campos deberán estar separados por ",". Cada clase será una línea.
Los datos que se deberán informar serán los siguientes en este orden:
Id, numero de clase, curso.
- Ejemplo de un fichero de importación de clases:

```
1,1,B
2,2,C
3,3,A
```

Figura 45.- Fichero importación clases.

- **Fichero importación alumnos:** el fichero deberá llamarse alumnos, sus campos deberán estar separados por “,” y cada alumno será una línea. Los datos que se deberán informar serán los siguientes en este orden:

Nombre, Apellidos, Fecha de nacimiento, id de clase.

Ejemplo de un fichero de importación de alumnos:

```
Pedro,Perez,,10-12-2001,1  
Sara,Sanz,,10-04-2002,1  
Luis,Ruiz,,09-02-2003,1  
Juan,Paredes,,08-01-2004,2  
Maria,Suarez,,23-08-2005,2  
Clara,Herranz,,14-09-2006,3
```

Figura 46.- Fichero importación Alumnos.

- **Fichero importación objetivos:** el fichero deberá llamarse objetivos, sus campos deberán estar separados por “,” y cada objetivo será una línea. Los datos que se deberán informar serán los siguientes en este orden:

Nombre, Descripción, Fecha, id de clase.

Ejemplo de un fichero de importación de objetivos:

```
Baloncesto,Botar balon,10-07-2018,1  
Balonmano,Pase,20-07-2018,2  
Hockey,Tiro,22-07-2018,3
```

Figura 47.- Fichero importación Objetivos.

- **Carga inicial desde la aplicación:** para crear alumnos, objetivos y clases desde la aplicación deberemos seleccionar las opciones de menú crear alumno, crear clase o crear objetivo. La creación de un alumno requiere que se haya creado previamente una clase, para poder realizar la asociación. Para su creación se deberán completar los datos que piden cada una de las pantallas siguientes:

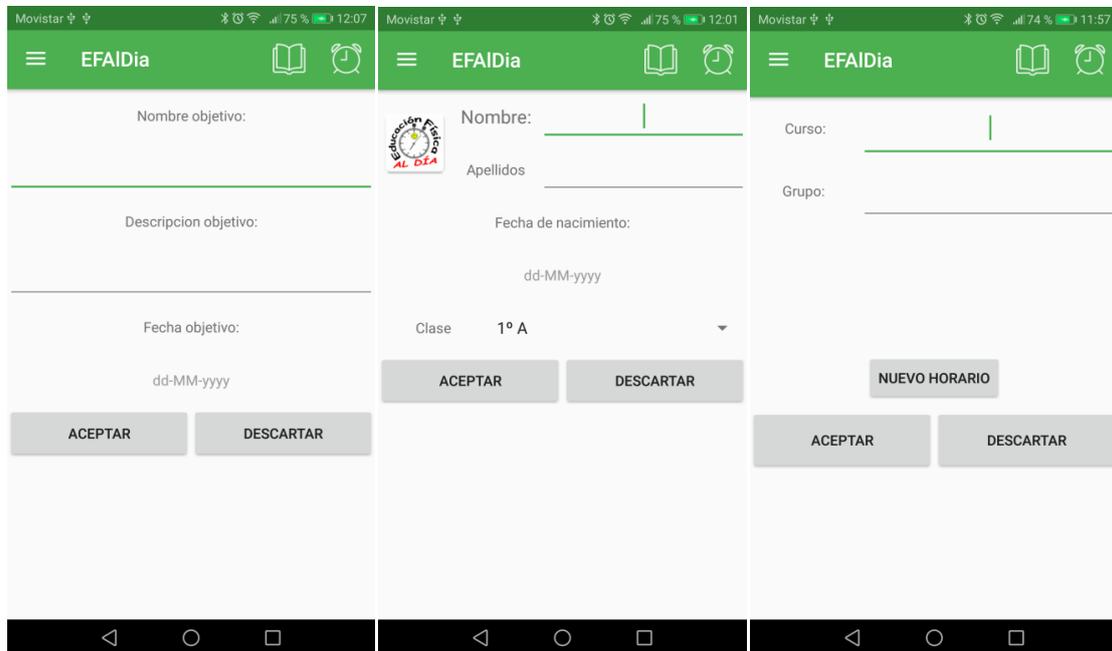


Figura 48.- Capturas de creación de objetivos, clases y alumnos.

- **Clase:** Una vez creados los datos, tendremos el calendario completo con las clases dibujadas. El acceso a una clase se podrá realizar de dos maneras. Seleccionando la clase en el horario o seleccionando el icono del menú superior “libro”. Si seleccionamos esta última opción, se abrirá una pantalla con todas las clases en la que deberemos seleccionar la clase a la que se desea acceder.

Una vez dentro de la ficha de la clase veremos dos botones:

- Un lápiz (edición): nos permitirá editar los datos de la clase.
- Un corredor (objetivos): nos mostrará la lista de objetivos para poder asociarlos a todos los alumnos de la clase.

También encontraremos por cada alumno de la lista un icono de borrado (papelera), que permitirá eliminar un alumno con todos sus registros.

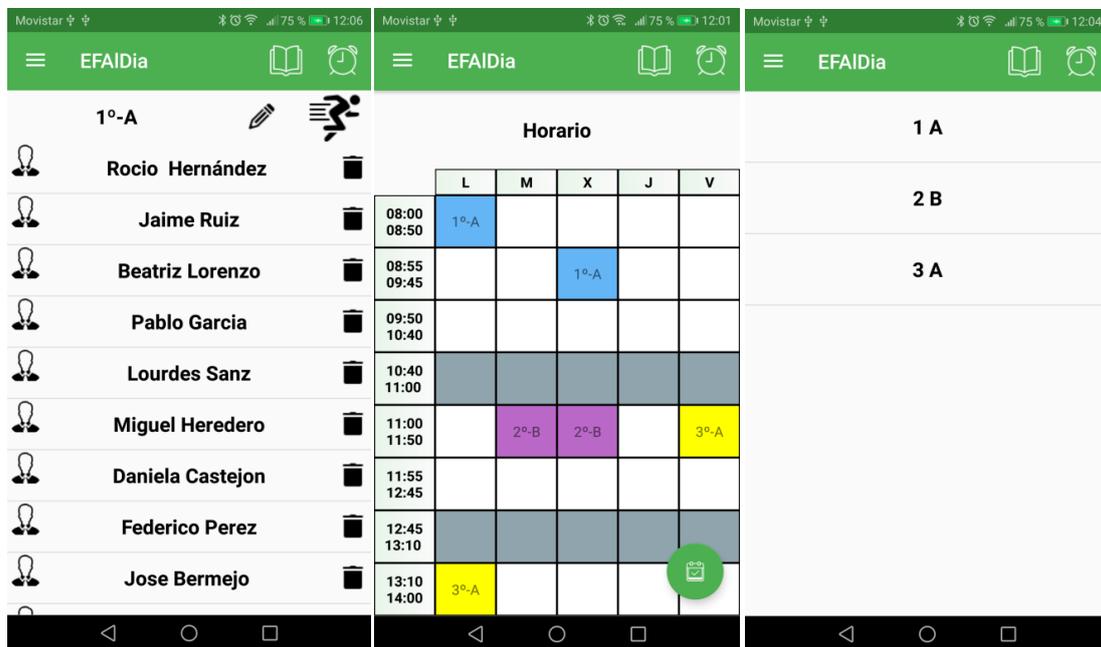


Figura 49.- Capturas de seguimiento de una clase.

- **Ver ficha alumno:** para acceder a la ficha de un alumno, se deberá seleccionar de la lista de alumnos pertenecientes a una determinada clase. Dentro de la ficha del alumno encontraremos sus datos personales, los objetivos que tiene asociados y sus anotaciones. Tanto los objetivos como las anotaciones disponen de un botón de borrado (papelera). Las anotaciones disponen de un botón de edición (lápiz), y los objetivos, de uno de aprobado/no aprobado.

Además también dispone de cuatro botones generales:

- Un corredor (objetivos): mostrará la lista de objetivos del alumno.
- Un lápiz (edición): nos permitirá editar los datos del alumno.
- Una grabadora: permitirá introducir anotaciones mediante audios transcritos.
- “Añadir nota”: permitirá introducir una anotación escrita mediante el teclado.

Si seleccionamos el icono del alumno podremos realizar una foto mediante la cámara para su mejor identificación.

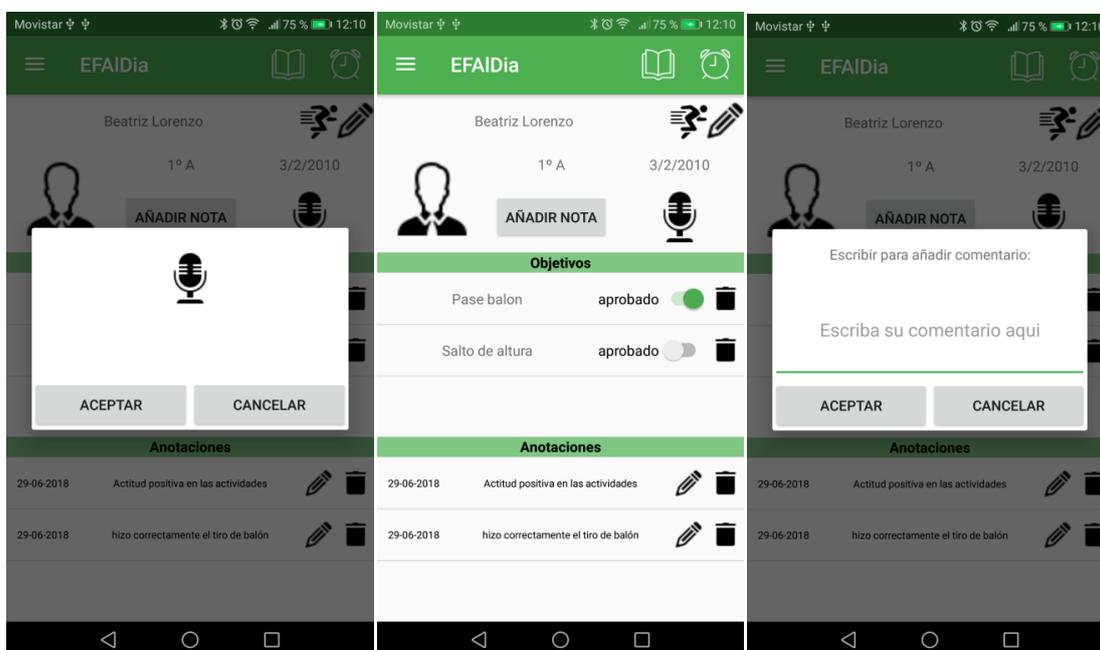


Figura 50.- Capturas de la pantalla ficha alumno.

- **Objetivos:** como se ha explicado anteriormente, los objetivos se pueden seleccionar de dos formas:
 - Desde una clase: seleccionado desde la clase, el objetivo se añadirá a la lista de objetivos de todos los alumnos que pertenecen a esa clase.
 - Desde un alumno: seleccionado desde un alumno, el objetivo se añadirá a la lista de objetivos de ese alumno.

Al seleccionar el botón “corredor”, tanto desde la clase como desde el alumno, se abrirá el siguiente dialogo, que permitirá seleccionar el objetivo que se quiere asociar.



Figura 51.- Captura de pantalla de asociación de objetivo.

- **Calendario:** para acceder al calendario se debe seleccionar el botón calendario de la pantalla principal (horario). Esta acción nos abrirá el calendario del mes en curso. Podemos observar que el día en el cual nos encontramos aparece coloreado en verde. A su vez, todas las alarmas que se hayan programado desde la aplicación se reflejarán mediante un punto verde debajo del día en el que se encuentran programadas.

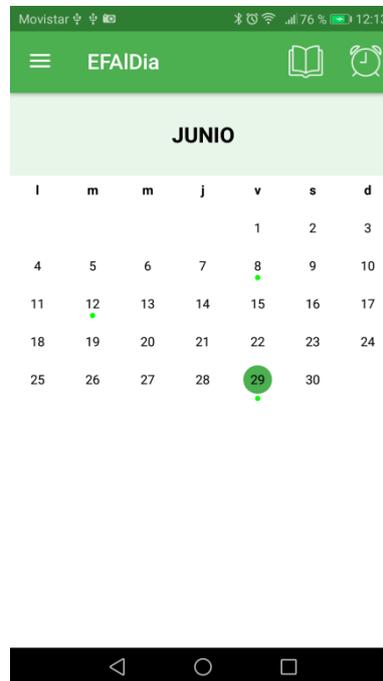


Figura 52.- Captura de pantalla del calendario.

- **Alarma:** Para la creación de una alarma se seleccionara el día deseado desde el calendario, esto abrirá un formulario con los datos necesarios para su creación. Las alarmas se pueden visualizar pulsando sobre el botón “reloj” del menú superior, que nos abrirá una lista con todas las alarmas. En ellas encontraremos
 - Icono de borrado (papelera): permitirá eliminar la alarma
 - Icono de activación/desactivación: permitirá modificar su estado.

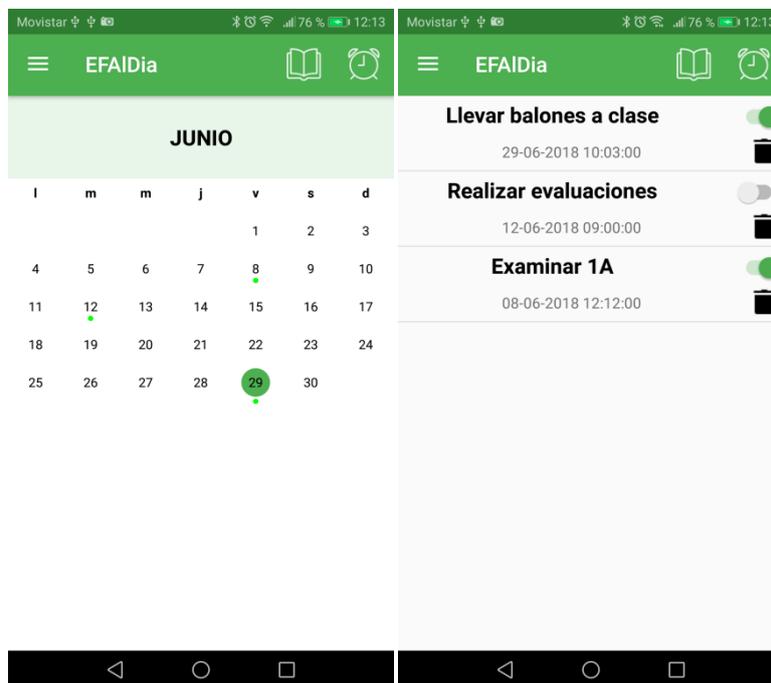


Figura 53.- Captura de pantalla de representación y lista de alarmas.

- **Menú:** Aparte de las opciones del menú que ya se han mencionado anteriormente, se disponen de tres opciones mas:
 - **Exportar todo:** permitirá exportar los datos de la aplicación a un fichero de texto, para poder examinarlos desde otro dispositivo.
 - **Cerrar sesión:** cerrará la sesión del usuario y redirigirá a la pantalla de login.
 - **Borrar todo:** esta opción del menú está pensada para cuando finalice el curso. Si el docente selecciona esta opción, se ejecutará un borrado completo de datos (alumnos, clases, comentarios, anotaciones y horarios). El sistema sólo conservará los datos de registro de la aplicación y redirigirá al docente a la pantalla de configuración de horario para que comience un nuevo curso.

SECCIÓN 4 – CONCLUSIONES

Capítulo 9 - Conclusiones y líneas futuras

9.1. Conclusiones

EFAIDia comienza como una propuesta de una alumna del grado de magisterio en educación física para solucionar el problema que tienen los docentes de esta materia a la hora de evaluar en continuo movimiento lo que les imposibilita tomar notas.

Este trabajo fue propuesto por los profesores José Vicente Álvarez y Belén Palop del Río que estuvieron presentes durante el proceso de análisis de requisitos y diseño de la aplicación. Posteriormente se incluyó a Aníbal Bregón para el apoyo tanto en el diseño como en el desarrollo.

Para consolidar la utilidad de este proyecto se reunió a varios docentes de educación física y se realizaron pruebas con un prototipo para testear la usabilidad del diseño y conocer las diferentes opiniones acerca de la aplicación móvil. Las pruebas fueron favorables, tras las opiniones decidimos la eliminación de los objetivos genéricos; puesto que los específicos se basan en ellos.

A lo largo de la implementación de la aplicación han surgido diversas dificultades ocasionadas por el desconocimiento de la aplicación Android Studio y el lenguaje Android.

Este proyecto me ha permitido conocer con más profundidad el mundo de las aplicaciones móviles, lenguajes, sistemas operativos, bases de datos y la comunidad que existe online para ayudar a solventar problemas de todo tipo.

Pero el mayor valor que me ha dado este proyecto ha sido el seguimiento de todo el proceso de desarrollo desde la petición de un “cliente”, reuniones para generar la interfaz de usuario y detectar necesidades, hasta su elaboración final.

9.2. Líneas futuras

Aunque el trabajo está finalizado y funciona correctamente, se requieren mejoras para su puesta en el mercado. A continuación se citan mejoras a corto plazo (tras realizarlas la aplicación será puesta en el play store de google) y largo plazo (para implementar en un futuro).

- A corto plazo:
 - Mayor flexibilidad en el horario: permitir adaptar la aplicación a cualquier tipo de horario, permitiendo incluir todas las franjas horarias deseadas y la posibilidad de ser aumentadas o disminuidas durante el curso. En este momento se permiten por motivos de visualización 8 franjas horarias, que se deberán cumplimentar al inicio de la aplicación.
 - Representación de objetivos en el calendario. Permitir desde el calendario visualizar si hay objetivos programados para ese día.

- A largo plazo:
 - Creación de gráficas que indiquen cómo ha evolucionado un alumno, que nos muestren una comparativa entre los objetivos aprobados y no aprobados.
 - Permitir generar la evaluación desde la propia ficha del alumno.
 - Permitir registrar la asistencia del alumno a clase.

Capítulo 10 - Bibliografía

- Documentación de la asignatura **Plataformas Software Móviles**.
Aníbal Bregón Bregón.
- Documentación de la asignatura **Modelado de Software de Sistemas de Información**.
Miguel A. Martínez Prieto
- **Android Developers**. Pagina web oficial para desarrolladores Android.
Último acceso: 01-06-2018
Disponible en: <https://developer.android.com/>.
- **Learning Android**. Marko Gargenta (2014).
Segunda edición. Editorial O'Reilly.
- **Foro Stack Overflow**. Pagina web de preguntas y respuestas para informáticos.
Último acceso: 28-05-2018
Disponible en: <https://stackoverflow.com>.
- **Comunidad Git Hub**. Plataforma para desarrolladores, posee un foro y herramientas de revisión de código.
Último acceso: 23-05-2018
Disponible en: <https://github.com/>.
- **Vogella**. Pagina web para la formación en tecnologías de la información, contiene tutoriales, libros y soporte.
Último acceso: 01-07-2018
Disponible en <http://www.vogella.com/>.

