

Ayuda al tutor.

*Aplicación web para la Gestión Tutorial de las Calificaciones en
la Educación Secundaria*

Autor: **Rafael Soriano Guasch**
Tutor: **Patricia Santos Rodríguez**
Profesor: **Tona Monjo Palau**

Estudios: **Grado de Ingeniería Informática**
Área: **Interacción Persona Ordenador**

Entrega: **01/2020**



Esta obra está sujeta a una licencia de

Reconocimiento-NoComercial-SinObraDerivada

[3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	<i>Ayuda al tutor. Aplicación web para centros de educación secundaria</i>
Nombre del autor:	<i>Rafael Soriano Guasch</i>
Nombre del colaborador/a docente:	<i>Patricia Santos Rodríguez</i>
Nombre del PRA:	<i>Tona Monjo Palau</i>
Fecha de entrega (mm/aaaa):	<i>01/2020</i>
Titulación o programa:	<i>Grado de Ingeniería Informática</i>
Área del Trabajo Final:	<i>Interacción Persona Ordenador IPO</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave:	<i>Educación secundaria, tutoría, automatización</i>

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo*

Desarrollo de una aplicación informática para uso en centros de educación secundaria, destinada a recoger las calificaciones de los alumnos, con el objetivo de que los profesores-tutores puedan disponer de información actualizada y completa acerca de sus alumnos, que les permita ejercer su función de seguimiento, control y comunicación con las familias, de manera óptima.

La facilidad de uso de la aplicación se considera fundamental. Con este objetivo, la metodología de diseño elegida ha sido DCU, diseño centrado en el usuario, y tal y como DCU propone, el usuario ha estado en todo momento en el centro del proceso. Se han realizado entrevistas en la fase de investigación, Card Sorting en la fase de elección de funcionalidades, etiquetado y estructura de menús (o sea, en la definición de la arquitectura de la información) y test de usabilidad en la fase de prototipado y posteriormente también sobre la primera versión de la aplicación real funcionando.

El resultado ha sido satisfactorio. Se dispone de una aplicación real, que cumple con las funcionalidades mínimas que se requerían y cuyo manejo es sencillo incluso para profesores con poca cualificación tecnológica.

Mi conclusión después del trabajo realizado sería: conocer a fondo la temática del proyecto es fundamental para poder llegar a un producto útil y sencillo, y si llegamos a tener un producto útil y

sencillo tendremos un producto muy bueno. La metodología DCU empuja al diseñador hacia el usuario, de forma que el diseñador acaba averiguando las necesidades reales del usuario y en consecuencia, diseñando el producto correcto.

Abstract (in English, 250 words or less):

Development of a computer application to be used in High School, aimed at collecting the qualifications of students, with the aim that teachers-tutors can have updated and complete information about their students, which allows them to exercise their role of monitoring, controlling and communicating with families, optimally.

The ease of use of the application is considered essential. With this objective, the chosen design methodology has been UCD, user-centered design, and as UCD proposes, the user has been at the center of the process at all times: interviews have been conducted in the research phase, Card Sorting in the phase of choosing functionalities, labeling and menu structure (that is, in the definition of the information architecture) and usability tests in the prototyping phase and subsequently also about the first version of the real application running.

The result has been satisfactory. A real application is available, which meets the minimum functionalities required and whose handling is simple, even for teachers with little technological skills.

My conclusion after this work would be, to know in depth the theme of the project is essential to reach a useful and simple product, and if we have a useful and simple product, we will have a very good product. The UDC methodology pushes the designer towards the user, so that the designer ends up finding out the real needs of the user and designing the correct product.

Índice

1.	Introducción	10
1.1.	Prefacio	10
1.2.	Descripción detallada	10
1.2.1.	Punto de partida	10
1.2.2.	Necesidad a cubrir y relevancia del tema	11
1.2.3.	Solución habitual en la actualidad	11
1.2.4.	Solución propuesta	11
1.2.5.	Desarrollo del producto	11
1.3.	Objetivos	12
1.3.1.	Objetivos generales	12
1.3.2.	Objetivos del producto	12
1.4.	Metodología y proceso de trabajo	13
1.4.1.	Fases del proceso de trabajo	13
1.5.	Planificación	15
1.6.	Estructura del resto del documento	16
2.	Análisis del mercado	17
2.1.	Importancia de la gestión de calificaciones dentro de la acción tutorial	17
2.1.1.	Opinión del público o usuario objetivo (profesorado)	18
2.2.	Herramientas informáticas para la gestión escolar	19
2.2.1.	Gestib	19
2.2.2.	SoftAula	19
2.2.3.	Clickedu	20
2.2.4.	Idoceo	20
2.2.5.	Additio	21
3.	Diseño	22
3.1.	Adaptación o desarrollo desde cero	22
3.2.	Arquitectura de la información	22

3.2.1.	Inventario de contenidos.....	23
3.2.2.	Card Sorting	25
3.2.3.	Árbol de contenidos (mapa del sitio).....	31
3.2.4.	Consideraciones personales	32
3.3.	Rutas rojas.....	32
3.4.	Escenarios.	33
3.4.1.	Escenario 1: Poner una nueva nota	33
3.4.2.	Escenario 2: Poner un comentario.....	33
3.4.3.	Escenario 3: Consultar notas de un alumno	33
3.5.	Diagramas de flujo de interacción	34
3.5.1.	Diagrama de flujo escenario 1: Poner una nueva nota.....	34
3.5.2.	Diagrama de flujo escenario 2: Poner un comentario	35
3.5.3.	Diagrama de flujo escenario 3: Consultar notas de un alumno.....	36
3.6.	Prototipado	37
3.6.1.	Sketching.....	37
3.6.2.	Wireframes	38
3.6.3.	Pruebas de usabilidad realizadas.....	42
3.6.4.	Nuevos wireframes después de las pruebas.....	44
3.6.5.	Siguiente paso.....	50
3.6.6.	Estudio del prototipo real implementado	51
3.6.7.	Resultados de las pruebas de usabilidad del prototipo real implementado	57
3.6.8.	Modificaciones tras las pruebas	59
4.	Implementación.....	61
4.1.	Lenguajes de programación y API utilizados.....	61
4.2.	Proceso de programación	61
4.2.1.	Fichero Profesores. Ejemplo de archivo perteneciente al modelo	62
4.2.2.	Fichero login. Ejemplo de archivo perteneciente a los controladores	63
4.2.3.	Fichero Identificación. Ejemplo de archivo perteneciente a las vistas	64
4.2.4.	Estructura de carpetas de la aplicación	65
4.3.	Requisitos de instalación.....	66

4.4. Instrucciones de instalación.....	66
5. Conclusiones.....	70
Bibliografía.....	71
Anexo 1: Hoja resumen de las entrevistas.....	73
Anexo 2: Modelo oficial de recogida de notas en el centro de referencia del TFG.....	77
Anexo 3: Lista de profesores cargados en la aplicación para poder hacer pruebas.....	80

Figuras y tablas

Índice de figuras

Ilustración 1 DCU, El usuario en el centro del proceso, provocando iteraciones y recibiendo sus resultados.....	16
Ilustración 2 Ejemplo de modelo empleado por los tutores para recoger información.....	17
Ilustración 3 Otro modelo de recogida de información por los tutores. Distinta disposición, pero misma filosofía.....	18
Ilustración 4 Captura de pantalla de la aplicación GESTIB.....	19
Ilustración 5 Cabecera con el logotipo de la web de SoftAula.....	20
Ilustración 6 Cabecera con el logotipo de la web de ClickEdu.....	20
Ilustración 7 Captura de pantalla de parte de la web de Idoceo.....	21
Ilustración 8 Captura de pantalla de la web de Additio.....	21
Ilustración 9 Vista gráfica de la base de datos con sus relaciones.....	23
Ilustración 10 Card Sorting del usuario 1.....	26
Ilustración 11 Card Sorting del usuario 2.....	27
Ilustración 12 Card Sorting del usuario 3.....	28
Ilustración 13 Card Sorting del usuario 4.....	29
Ilustración 14 Card Sorting del usuario 5.....	30
Ilustración 15 Diagrama de flujo del escenario 1.....	34
Ilustración 16: Diagrama de flujo del escenario 2.....	35
Ilustración 17 Diagrama de flujo del escenario 3.....	36
Ilustración 18 Sketching inicial de poner notas.....	37
Ilustración 19 Sketching inicial del índice.....	37
Ilustración 20 Sketching de Ver medias.....	38
Ilustración 21 Sketching inicial de Ver notas.....	38
Ilustración 22 Primeros wireframes. Menú principal.....	39
Ilustración 23 Primeros wireframes. Pantalla Inicio.....	39
Ilustración 24 Primeros wireframes. Modificar notas.....	40
Ilustración 25 Primeros wireframes. Poner notas.....	40
Ilustración 26 Primeros wireframes. Ver medias.....	41
Ilustración 27 Primeros wireframes. Notas de un alumno.....	41
Ilustración 28 Primeros wireframes. Notas de una asignatura.....	42
Ilustración 29 Wireframes mejorados. Pantalla de menú principal.....	45
Ilustración 30 Wireframes mejorados. Pantalla de inicio.....	45
Ilustración 31 Wireframes mejorados. Modificar nota.....	46
Ilustración 32 Wireframes mejorados. Poner nueva nota.....	46
Ilustración 33 Wireframes mejorados. Ver notas de una asignatura.....	47
Ilustración 34 Wireframes mejorados. Notas de un alumno.....	47

Ilustración 35 Wireframes mejorados. Medias de asignatura.....	48
Ilustración 36 Wireframes mejorados. Modificar comentario.....	49
Ilustración 37 Wireframes mejorados. Poner comentario.....	49
Ilustración 38 Wireframes mejorados. Ver comentarios.....	50
Ilustración 39 Ejemplo de menú no aconsejado para su implementación.....	51
Ilustración 40 Pantalla de login.....	53
Ilustración 41 Pantalla de inicio. Menú principal de la aplicación.....	54
Ilustración 42 Pantalla Poner notas.....	54
Ilustración 43 Pantalla Poner comentarios.....	55
Ilustración 44 Pantalla Seguimiento alumnos.....	56
Ilustración 45 Representación gráfica del patrón Modelo-Vista-Controlador.....	62
Ilustración 46 Estructura de carpetas de la aplicación reflejando la estructura del patrón MVC.....	65
Ilustración 47 Ficheros de la aplicación.....	66
Ilustración 48 Página raíz (localhost) en instalación local de un WAMP.....	67
Ilustración 49 Importando la base de datos.....	68
Ilustración 50 Modelo oficial de recogida de notas. Hoja 1.....	77
Ilustración 51 Modelo oficial de recogida de notas. Hoja 2.....	78
Ilustración 52 Modelo oficial de recogida de notas. Hoja 3.....	79

Índice de tablas

Tabla 1 Diagrama de Gantt del TFG.....	15
Tabla 2 Relación de las tablas de la base de datos con sus respectivos campos.....	23
Tabla 3 Tabla de contenidos estructurados por niveles.....	24
Tabla 4 Tarjetas para el Card Sorting. Modelo listo para imprimir y recortar.....	26
Tabla 5 Listado de usuarios facilitados para pruebas de usabilidad con la aplicación real.....	53
Tabla 6 Resumen de la información recogida en las pruebas de usabilidad.....	58
Tabla 7 Listado de usuarios (profesores) válidos para probar la aplicación.....	80

1. Introducción

1.1. Prefacio

El cambio organizativo que se da en nuestro sistema educativo al cambiar de los estudios de primaria a los de secundaria, hace que los alumnos pasen de tener un profesor principal (el tutor) al cual ven varias horas al día y que les imparte las principales asignaturas, a tener un profesor por asignatura, que cambia cada hora y que desconoce los resultados de los alumnos en las otras asignaturas. Esta estructura provoca que, a pesar de las reuniones de coordinación, el tutor de un grupo de secundaria carezca de una información completa y actualizada, y a veces tarde en detectar la bajada de rendimiento de un determinado alumno, y en consecuencia, también tardará en contactar con la familia y en tomar medidas para corregir esta situación.

El autor de este trabajo trabaja desde hace años en un IES (instituto de enseñanza secundaria) y a lo largo de este periodo ha ejercido de tutor en varias ocasiones. Fruto de esta experiencia se ha llegado a la conclusión de que mantener una información al día de los resultados de los alumnos del grupo permite actuar antes y solucionar determinados temas de una manera mucho más sencilla y eficaz que cuando la intervención es más tardía. El inconveniente es que no es fácil recoger esta información. Un grupo, una clase, tiene de media 10 asignaturas, es decir, 10 profesores, cada uno de ellos con sus notas. Además, los profesores, obviamente dan clases a otros grupos, con sus respectivos tutores, con lo cual, si todos los tutores deciden solicitar de manera frecuente información de toda la clase a los profesores se generará una sobrecarga considerable de papeleo. Es evidente que la coordinación manual de esta situación es complicada y poco eficiente. La consecuencia lógica de esta circunstancia es que en la práctica el tutor sólo pedirá información de un alumno cuando tenga la visita de sus padres y si esto es así, al no hacer un seguimiento más exhaustivo se puede perder la posibilidad de actuar en las etapas iniciales del problema.

Se necesita, en definitiva, que el tutor tenga a su alcance la máxima información de sus alumnos y lo más actualizada posible. Para conseguirlo, en este TFG se pretende desarrollar una aplicación informática que permita a los profesores introducir las notas de los alumnos, y a los tutores consultarlas, corrigiendo así la carencia de información sobre los estudiantes que la estructura de "una asignatura/un profesor" presenta en la educación secundaria frente a la estructura organizativa menos dispersa que se da en la educación primaria.

1.2. Descripción detallada

1.2.1. Punto de partida

Tal como se ha descrito en el prefacio anterior, en la educación secundaria partimos de una estructura organizativa en la cual tenemos un profesor para cada asignatura. Esta estructura favorece la especialización, pero dificulta la coordinación y la comunicación, al haber más partes implicadas. Esto se da, además, en una etapa de la vida del alumno, la adolescencia, en la que los cambios en actitudes

y hábitos de comportamiento son habituales, por cuanto conviene más aún si cabe hacer un seguimiento cercano de su proceder y resultados.

1.2.2. Necesidad a cubrir y relevancia del tema

Una situación como esta plantea al tutor la necesidad de disponer de la máxima información posible, que le permita detectar de manera temprana situaciones como falta de motivación, comportamientos disruptivos, dificultades con determinadas asignaturas, etc. y actuar en consecuencia para corregir el problema.

Tenemos por tanto una estructura organizativa en la Educación Secundaria que por su propia definición hace difícil disponer de información actualizada, situado dentro de un tema sensible como es la educación de adolescentes y que lo convierte por tanto en un problema de especial relevancia. ¿Cómo lo resolvemos?

1.2.3. Solución habitual en la actualidad

Para resolver esta problemática en la actualidad, lo que se hace en la mayoría de los centros de educación secundaria cuando un tutor quiere tener información detallada sobre un alumno (normalmente porque espera la visita de sus padres) es pedir al resto de los profesores del grupo que pasen una recopilación de sus calificaciones y si lo consideran necesario algún comentario sobre la actitud del alumno en clase. El proceso normalmente consiste simplemente en comentarlo directamente de viva voz o en pasarse una hoja, una plantilla, entre los profesores afectados para rellenar las notas y comentarios pertinentes al alumno en cuestión.

1.2.4. Solución propuesta

Se propone desarrollar una aplicación que permita mantener un repositorio de datos común, donde diferentes profesores apunten las notas de sus alumnos y el tutor pueda consultarlas. Esta solución aportará a los tutores un flujo de información actualizada referida a los alumnos que les permitirá actuar de manera temprana cuando se produzcan situaciones que necesiten ser corregidas, ya sean estas conductuales o académicas.

1.2.5. Desarrollo del producto

Para desarrollar el producto adecuado, vamos a estudiar a los usuarios: sus necesidades, enfoques y maneras de trabajar. De acuerdo con este planteamiento, centraremos nuestros esfuerzos en dos vertientes:

- El diseño, que nos permitirá, a partir de las necesidades y el enfoque que exprese el usuario, definir de manera precisa las funcionalidades, la estructura y el etiquetado adecuados de la aplicación.
- La disponibilidad. Un profesor de secundaria reparte su jornada laboral entre diversas aulas, la sala de profesores, el patio, su propia casa, ... En este contexto es fácil deducir que la manera de trabajar de un profesor hace que pueda poner notas desde casa con su ordenador personal, desde la sala de profesores quizás con una Tablet, o incluso en una guardia de patio

con su móvil, por lo cual sería deseable que la aplicación estuviera al alcance del usuario desde cualquier dispositivo.

Teniendo en cuenta todo esto, vamos a llevar a cabo un proceso de diseño aplicando DCU¹, que nos permitirá llegar a una definición concreta del producto que realmente necesita el usuario y después procederemos a implementar este diseño en una aplicación web con diseño responsive² o adaptativo que nos garantizará que el usuario podrá acceder desde cualquier dispositivo con el único requisito de que tenga acceso a Internet.

1.3. Objetivos

1.3.1. Objetivos generales

Los objetivos generales de este trabajo son:

- Integrar los conocimientos adquiridos a lo largo de toda la carrera y utilizarlos.
- Llevar a cabo un desarrollo completo que permita una perspectiva global, diferente de la que ofrece cada asignatura por separado.
- Hacer hincapié a lo largo del proyecto en la interacción entre persona y ordenador, aplicando la metodología DCU durante todo el proceso

1.3.2. Objetivos del producto

Los objetivos que debe alcanzar el sistema que se pretende desarrollar son:

- Implementar un sistema donde los profesores autorizados puedan insertar notas.
- Obtener a partir de estas notas informaciones de resumen y de tendencias, básicamente cálculo de distintas medias.
- Permitir solo al profesor-tutor consultar las notas introducidas y las informaciones obtenidas de estas notas.
- Permitir a los profesores (no tutores) ver la misma información, pero referida sólo a su asignatura y en algunos casos, una información de resumen.
- Facilitar tanto como sea posible la usabilidad del sistema. El objetivo es hacer una herramienta que quite trabajo, no que lo añada.

¹ Diseño Centrado en el Usuario

² Filosofía de diseño y desarrollo cuyo objetivo es adaptar la apariencia de las páginas web al dispositivo que se esté utilizando para visitarlas. Wikipedia [19]

1.4. Metodología y proceso de trabajo

Este proyecto busca ante todo la facilidad de uso de la aplicación, es decir, conseguir la mejor Usabilidad o UX³ que sea posible. Para conseguir este objetivo se seguirá la metodología DCU, que estipula que las decisiones están dirigidas por y hacia el usuario, es decir, el usuario está siempre en el centro del proceso⁴.

El trabajo se ha llevado a cabo en las fases que se especifican a continuación. No obstante, cabe destacar que de acuerdo con la metodología DCU seguida, el proceso es iterativo, de manera que las fases no se realizan completas de forma secuencial, sino que guiados por la información obtenida de los usuarios se pueden producir cambios que nos lleven de nuevo a una fase anterior para introducir modificaciones y, o mejoras.

1.4.1. Fases del proceso de trabajo

- Investigación y definición: En esta fase, por una parte se ha estudiado la oferta que hay en el mercado de productos con funcionalidades similares (análisis de mercado). El objetivo era decidir si se podía diseñar una mejora sobre algún producto ya existente en lugar de crear un producto nuevo desde cero. También se pretendía conocer de qué manera se satisfacían las necesidades que plantea nuestro proyecto por parte de la competencia y así aprender de su experiencia. Por otra parte, siguiendo la metodología DCU, en esta fase se han tenido las primeras entrevistas con usuarios, con el objetivo de corroborar o refutar las ideas de las que partimos, detectar nuevas funcionalidades y en general tener una idea más precisa de lo que se pretende desarrollar. Como se verá más adelante en este trabajo, fruto de esta fase y en concreto de lo obtenido de las entrevistas con usuarios se ha añadido al proyecto la posibilidad de introducir comentarios (además de notas) y se confirmó la necesidad de desarrollar un producto multiplataforma.
- Diseño: Esta es la fase principal de todo el proyecto. Todas las iteraciones que se han hecho nos devolvían siempre a este punto. En esta fase se parte del conocimiento adquirido en la fase anterior para al final llegar a un prototipo de un producto concreto y evaluable por parte del usuario. Continuando con la aplicación de la metodología DCU, para llevar a cabo el diseño se han realizado las siguientes tareas:
 - Definición de la arquitectura de la información de la aplicación. Mediante la arquitectura de la información se pretende estructurar los contenidos de la aplicación que se está desarrollando, establecer un etiquetado (labeling) que facilite la comprensión al futuro usuario y definir el sistema de navegación que se va a emplear para que el usuario pueda moverse por la aplicación sin problemas. Para llevar a cabo la arquitectura se han realizado, en primer lugar, un inventario de la información que maneja la aplicación, escrito en forma de tabla de contenidos. A partir de esta tabla se

³ User Experience. Sinónimo de Usabilidad

⁴ Se puede leer una explicación muy clara de lo que son UX y DCU en "Metodología DCU" [11]

ha preparado un card sorting con usuarios, y de resultados de este se ha decidido cómo será el mapa del sitio o árbol de contenidos y una propuesta de menú.

- Establecimiento de unas rutas rojas (según se describen en The Fable of the User-Centered Designer [18]). Son las funcionalidades más importantes de la aplicación, en nuestro caso son: Poner notas, Poner comentarios y Ver notas, ya que son las funcionalidades que garantizan que al menos lo que ya se hacía de manera manual se sigue haciendo.
- Descripción de escenarios y diagramas de flujo para las citadas rutas rojas, donde se describe de manera detallada cómo sería la realización de esas tareas.
- Creación de los prototipos. Teniendo en cuenta toda la información recogida en las etapas anteriores, se empiezan a dibujar las pantallas de la aplicación.

Siempre en el DCU, pero sobre todo en esta etapa, se involucrará al usuario para que evalúe los resultados que se van obteniendo, y se iterará la versión obtenida si consideramos conveniente aplicar sus propuestas.

- Evaluación. Como ya se ha visto al describir las etapas anteriores, esta no es una fase aislada y separada en el tiempo de las otras, sino que se ha realizado de manera continua a lo largo de todo el proceso: siempre que se ha obtenido algún resultado se ha sometido a evaluación, y siempre para esta evaluación se ha tenido en cuenta a los usuarios.
- Implementación. En esta fase, partiendo de los prototipos probados con los usuarios se ha llevado a cabo la programación. Se han utilizado HTML, CSS, JavaScript y PHP como lenguajes; Bootstrap 4 [2] como framework de HTML, CSS y JavaScript para facilitar la programación web adaptativa y JQuery [14] como librería JavaScript. La aplicación se ha estructurado según el patrón Modelo-Vista-Controlador, separando los ficheros en estas tres categorías según se explicará más adelante. Una vez disponible la versión beta del producto, se ha puesto a disposición de los usuarios para una última evaluación de la cual no han surgido cambios relevantes.

1.5. Planificación

Siguiendo el ritmo de entregas marcado por las PAC de la asignatura, los trabajos necesarios para culminar este proyecto y su planning de ejecución han sido:

Tareas	Inicio	Fin	18/09/2019	25/09/2019	30/09/2019	01/10/2019	08/10/2019	14/10/2019	15/10/2019	26/10/2019	02/11/2019	09/11/2019	16/11/2019	23/11/2019	30/11/2019	07/12/2019	11/12/2019	21/12/2019	28/12/2019	04/01/2020	09/01/2020	16/01/2020	20/01/2020
PAC 1 Definición del TFG	18-sep	30-sep																					
Elección del tema	18-sep	25-sep																					
Redacción de la propuesta	26-sep	30-sep																					
PAC 2 Análisis de mercado	01-oct	14-oct																					
Situación actual	01-oct	05-oct																					
Oferta de productos similares	06-oct	14-oct																					
Entrevistas a profesores	01-oct	14-oct																					
PAC 3 Diseño	15-oct	11-dic																					
Arquitectura de la información	15-oct	02-nov																					
Evaluación con usuarios (Card Sorting)	27-oct	02-nov																					
Prototipado sketching	03-nov	12-nov																					
Prototipado wireframing I	13-nov	23-nov																					
Evaluación con usuarios (Usabilidad)	24-nov	07-dic																					
Prototipado wireframing II	08-dic	11-dic																					
PAC 4 Confección de la memoria	15-oct	09-ene																					
Implementación programa real	08-oct	20-ene																					
Estudio tecnologías de implementación	08-oct	25-oct																					
Nociones sobre Web responsive	15-oct	05-nov																					
Diseño de la base de datos	22-oct	26-oct																					
Estudio de leng. de programación	06-nov	20-ene																					
Programación de la aplicación	30-nov	20-ene																					
PAC5 Presentación y defensa del TFG	16-ene	20-ene																					

Tabla 1 Diagrama de Gantt del TFG

1.6. Estructura del resto del documento

La estructura del trabajo viene marcada por la aplicación de la metodología DCU, que propone trabajar en tres grandes áreas: investigación, diseño y evaluación.

En el **capítulo 2** se describe como se ha realizado el proceso de investigación de mercado, con la finalidad de conocer mejor el dominio del problema que se pretende resolver.

Con todo el conocimiento recogido en la fase de investigación estaremos en condiciones de, en el **capítulo 3**, iniciar el diseño de la aplicación. Por la forma de trabajar mediante iteraciones, que conlleva la metodología DCU, en este capítulo 3 se ha incluido también la fase de evaluación. Gráficamente lo podríamos representar así:

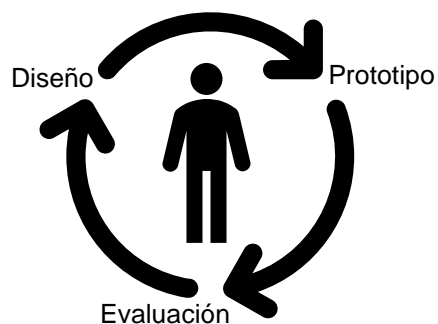


Ilustración 1 DCU, El usuario en el centro del proceso, provocando iteraciones y recibiendo sus resultados

Nuestro último prototipo será una implementación funcional completa, que podríamos ver como la versión inicial o beta del producto. En el **capítulo 4** se explica cómo ha sido esta fase de implementación.

Para terminar, en el **capítulo 5**, las conclusiones, se describe como ha sido la experiencia y el aprendizaje que se desprende de ella, junto con alguna apreciación particular.

2. Análisis del mercado.

El público objetivo de nuestro producto es el profesor de educación secundaria. Interesa estudiar el mercado para, en primer lugar, tratar de averiguar cuáles son las recomendaciones, las pautas más habitualmente seguidas en cuanto a gestionar la tutoría de un curso, en particular en la gestión de la información de las calificaciones de los alumnos, a fin de medir la utilidad de la aplicación que se está desarrollando. En segundo lugar, conocer la oferta de productos similares, para lo cual, se ha procedido a encontrar y clasificar herramientas informáticas dirigidas a la gestión de la labor docente de profesores y centros escolares, que dispongan de opciones que permitan llevar a cabo esta tarea de gestión tutorial o por lo menos, que la apoyen (también haciendo especial incidencia en las calificaciones).

2.1. Importancia de la gestión de calificaciones dentro de la acción tutorial

¿Realmente la automatización de la gestión de calificaciones que se pretende desarrollar satisface una necesidad existente dentro del ámbito de la educación secundaria? Para empezar, ¿cómo lo hacen actualmente los tutores? En el libro "GUÍA DEL TUTOR DE SECUNDARIA" de Claudio Castilla [5] encontramos una propuesta que se podría considerar como habitual:

CASTILLA ROMERO, CLAUDIO (2012). Guía del Tutor de Secundaria. Murcia: Consejería de Educación, Formación y Empleo.

Hoja de recogida de información

Alumno/a: _____		Curso: _____		Fecha: _____	
MATERIA: _____	Prof.: _____	MATERIA: _____	Prof.: _____		
Asistencia/Actitud.		Asistencia/Actitud.			
Aprendizaje/Notas.		Aprendizaje/Notas.			
Trabajo en casa.		Trabajo en casa.			
MATERIA: _____	Prof.: _____	MATERIA: _____	Prof.: _____		
Asistencia/Actitud.		Asistencia/Actitud.			
Aprendizaje/Notas.		Aprendizaje/Notas.			
Trabajo en casa.		Trabajo en casa.			
MATERIA: _____	Prof.: _____	MATERIA: _____	Prof.: _____		
Asistencia/Actitud.		Asistencia/Actitud.			

Ilustración 2 Ejemplo de modelo empleado por los tutores para recoger información

Es fácil deducir el proceso que se seguirá: el tutor imprimirá la hoja y la irá pasando al resto de profesores que imparten clases a su grupo. De media un curso de secundaria tiene diez asignaturas, normalmente cada una con su profesor. Además, cada profesor tiene su horario, con lo que puede suceder que el tutor no coincida por los pasillos del centro o la sala de profesores con alguno de los docentes a corto plazo. Finalmente, por experiencia sabemos que la reunión con los padres se concertará con dos, tres días de antelación. Con estas premisas, lo más probable es que la recogida de información no sea ni de lejos exhaustiva.

Otro ejemplo de propuesta actual se encuentra en la página web <https://www.orientacionandujar.es/> [13]. Es interesante destacar que se trata de un blog hecho por profesores y orientadores de secundaria y basado en su experiencia, lo cual le otorga si cabe, una mayor credibilidad.

INFORME DEL PROFESORADO AL TUTOR/A.

Por inminente entrevista con la familia, necesito información del alumno/a

curso _____.

Materia	Asistencia y comportamiento en clase	Realización de tareas, trabajos, lecturas, etc.	Participación en clase e integración grupal	Dificultades. Necesidad de estudio, apoyo	Otros datos para comentar con la familia
ASIGNATURA 1					
ASIGNATURA 2					
ASIGNATURA 3					
ASIGNATURA 4					
ASIGNATURA 5					

Ilustración 3 Otro modelo de recogida de información por los tutores. Distinta disposición, pero misma filosofía

A pesar de tener ítems diferentes, es evidente que el proceso asociado a este documento tendrá una mecánica muy similar al anterior.

2.1.1. Opinión del público o usuario objetivo (profesorado)

Finalmente, aprovechando la circunstancia de que el autor de este TFG trabaja como profesor en un instituto de enseñanza secundaria, se han recabado las opiniones de sus compañeros de claustro. El resumen de las entrevistas realizadas se adjunta al final de este documento como Anexo 1. Así mismo, para reflejar también la forma de hacer del centro se adjunta como Anexo 2 el modelo oficial que se emplea para este cometido.

De esta información recabada en el centro se concluye que:

- El procedimiento principal se basa en rellenar la hoja establecida.
- En muchos casos, sobre todo si hay cercanía a nivel personal, se utiliza el WhatsApp, pese a estar desaconsejado por temas de protección de datos.
- Algunos tutores, aprovechan la hora de tutoría para solicitar a todos los alumnos las notas obtenidas a lo largo de la semana. De esta manera mantienen un registro bastante actualizado, aunque sujeto a ciertas reservas en cuanto a su veracidad (en ocasiones, el adolescente puede “mejorar” una nota para evitar la tan temida llamada a casa).

Los documentos mostrados y otros similares que se han consultado, junto con las entrevistas realizadas a los docentes y la experiencia personal del autor como profesor y tutor, muestran que,

dentro de la gestión tutorial, la información de las notas del alumno es una parte ineludible e importante del conjunto de datos que necesita el tutor para llevar a cabo su trabajo. Igualmente, queda patente que el proceso de tratamiento de dicha información es manifiestamente mejorable.

2.2. Herramientas informáticas para la gestión escolar

Así pues, aceptando que la funcionalidad en estudio es importante dentro del total del trabajo de un tutor, ¿qué herramientas informáticas se encuentran en el mercado que apoyen esta tarea?

Obviamente, tan solo listar de manera exhaustiva todas las aplicaciones de gestión escolar es un trabajo casi interminable y fuera del alcance del autor de este TFG. Se verá pues, una colección de aplicaciones representativas, con sus diferentes características y opciones, y que nos permitirán tener una idea aproximada de lo que ofrece el mercado hoy en día.

2.2.1. Gestib

Aplicación para la gestión educativa de las Islas Baleares [10]. Es la herramienta oficial de la comunidad autónoma. Todos los centros de la comunidad están obligados a emplearla. Funcionalidades para matrícula de alumnos, gestión de sus datos académicos, personales e incluso médicos. Sin embargo, en lo que respecta a las calificaciones, limitada a notas finales de evaluación (primera, segunda y final), falta por tanto, la posibilidad de hacer un seguimiento más detallado en lo que a notas se refiere.



Ilustración 4 Captura de pantalla de la aplicación GESTIB

2.2.2. SoftAula

Aplicación muy completa, aunque no queda claro en su web si dispone de la funcionalidad que se busca. La descripción que se hace de ella como ERP y CRM evidencia que su orientación es de ámbito global, para todo el centro, saliéndose del alcance de este estudio. Permite un nivel de concreción de notas superior al del Gestib, pero no parece ágil para el uso entre iguales (compartir información entre profesores) que se quiere conseguir. Tiene una versión en la nube, de acceso web, pero la versión que parecería más cercana para lo que se pretende en este proyecto es la versión Server, con la limitación

de que solo está disponible en entornos MS-Windows. Se puede encontrar información de SoftAula en su web: <https://softaula.net/> [17]



Ilustración 5 Cabecera con el logotipo de la web de SoftAula

2.2.3. Clickedu

Se puede considerar esta aplicación como el producto más completo y adecuado de los vistos a lo largo de este proceso de búsqueda. Refiriéndose siempre a la información que se facilita en la web, se ha visto que el programa contempla y da importancia a la comunicación entre profesores y familias, que es la piedra angular del espíritu de este proyecto. Obviamente, va mucho más allá y abarca todo un conjunto de posibilidades que este TFG ni pretende ni puede alcanzar. Por razones de tiempo y de planificación no se solicitará la demo de esta aplicación (existe la posibilidad en la web), ni se hará un estudio más profundo de lo que realmente ofrece, pero queda anotado en la agenda del autor de este trabajo como una intención de futuro.

Se trata de una aplicación web que permite entre otras muchas cosas dar de alta a los padres de los alumnos y, si el profesor introduce las notas, enviar comunicaciones personalizadas de manera casi inmediata. Citando literalmente la publicidad de la web: *“Desde que una escuela tiene Clickedu, las familias pasan a estar informadas de aspectos tales como: noticias, álbumes de fotos, notas, exámenes, información del día a día en clase, calendarios online, mensajes de los tutores, visualización de fechas para concertar entrevistas, descarga online de los boletines de notas, gestión de las actividades extraescolares, menús y comedor...”*

La dirección de la página web de la aplicación es <https://clickartedu.com/inicio-plataforma-colegios.html> [15]

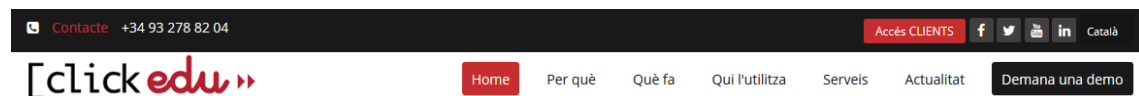


Ilustración 6 Cabecera con el logotipo de la web de ClickEdu

2.2.4. Idoceo

Viene definida como “cuaderno del profesor” y como tal, cumple perfectamente las expectativas. El problema que presenta es que no tiene previsto compartir notas (el profesor con el tutor) con lo que no da soporte a la funcionalidad que se busca en este proyecto. Una limitación importante es que se trata de una aplicación exclusiva para iPad.

Se trata de una aplicación de alta implantación entre el profesorado, lo que da idea de su calidad y utilidad. Toda la información de esta aplicación puede encontrarse en su web: www.idoceo.es [12]



Ilustración 7 Captura de pantalla de parte de la web de Idoceo

2.2.5. Additio

Hasta hace poco era una aplicación muy parecida a la anterior, con las mismas características en lo que respecta a funcionalidades, y también con fuerte implantación entre el profesorado. A diferencia de Idoceo, esta aplicación era únicamente para entornos Android. Actualmente también está disponible para IOS, e incluso ofrecen una versión web para centros. A pesar de esto, en la información que se ofrece no hacen una referencia explícita a una compartición sencilla de notas entre profesores.

La web del fabricante es www.additioapp.com [1]



Ilustración 8 Captura de pantalla de la web de Additio

3. Diseño

3.1. Adaptación o desarrollo desde cero

Llegados a este punto ya se tiene claro el producto que se pretende desarrollar. Para obtenerlo, una posibilidad a valorar es añadir la funcionalidad deseada a una aplicación ya existente. En este caso, se consideró el entorno Moodle. Como pros de esta opción está el hecho de que es una herramienta de código abierto y, además existe una gran comunidad que le da soporte. También, el hecho de que muchos institutos de secundaria ya usan su propio Moodle, con lo cual en estos casos el proceso de implantación de la herramienta quedaría eliminado.

Para explorar esta posibilidad se solicitó la colaboración de un pequeño grupo de profesores, usuarios habituales de Moodle, y se llevó a cabo una lluvia de ideas (brainstorming), fruto de la cual finalmente se obtuvieron dos propuestas:

- Crear un curso y dentro de este, cada asignatura será un tema. Los profesores crearán una tarea nueva dentro de su asignatura por cada nota.
- Crear una categoría para cada grupo. Dentro de la categoría crear para cada asignatura un curso, que gestionará cada profesor de asignatura. Desarrollar una nueva funcionalidad de Moodle (básicamente una exportación de notas a Excel) que permita obtener las notas de toda la categoría en un único documento.

Teniendo en cuenta los pros, y también las propuestas presentadas, se debe valorar como contras para la opción de hacer una implantación sobre el Moodle el hecho de que la simplicidad de uso es un factor básico para este proyecto. Cualquier modificación hecha a Moodle hereda la potencia, pero también la complejidad del sistema. Para la primera propuesta, introducir una nota implica entrar al Moodle, crear una nueva tarea y picar las calificaciones. No es un proceso sencillo. Por lo que respecta a la segunda propuesta, el procedimiento es igualmente complicado e inasumible si tenemos en cuenta el perfil informático de un buen número de profesores en casi cualquier centro.

Así pues, se descartó la opción de adaptar el Moodle a nuestras necesidades y se ha optado por hacer un desarrollo completo desde cero.

3.2. Arquitectura de la información

Mediante la arquitectura de la información se pretende estructurar los contenidos de la aplicación que se está desarrollando, establecer un etiquetado (labeling) y definir el sistema de navegación que se va a emplear.

De acuerdo con esto, en primer lugar, se ha de llevar a cabo un inventario detallando cuál es la información que manejará la aplicación.

3.2.1. Inventario de contenidos

Con el conocimiento del problema que hemos acumulado en las fases anteriores estamos ya en condiciones de definir la base de datos que tendrá la aplicación:

Tabla	Campos
Profesor	id, nombre, apellidos, email, password, estado
Grupo	id, nombre, id_profesor
Asignatura	id, nombre, id_grupo, id_profesor
Alumnos	id, nombre, apellidos, id_grupo
Exámenes	id, nombre, fecha, evaluación, id_asignatura
Notas	id, valor, id_alumno, id_examen
Comentarios	id, texto, fecha, id_profesor, id_alumno

Tabla 2 Relación de las tablas de la base de datos con sus respectivos campos

Gráficamente:

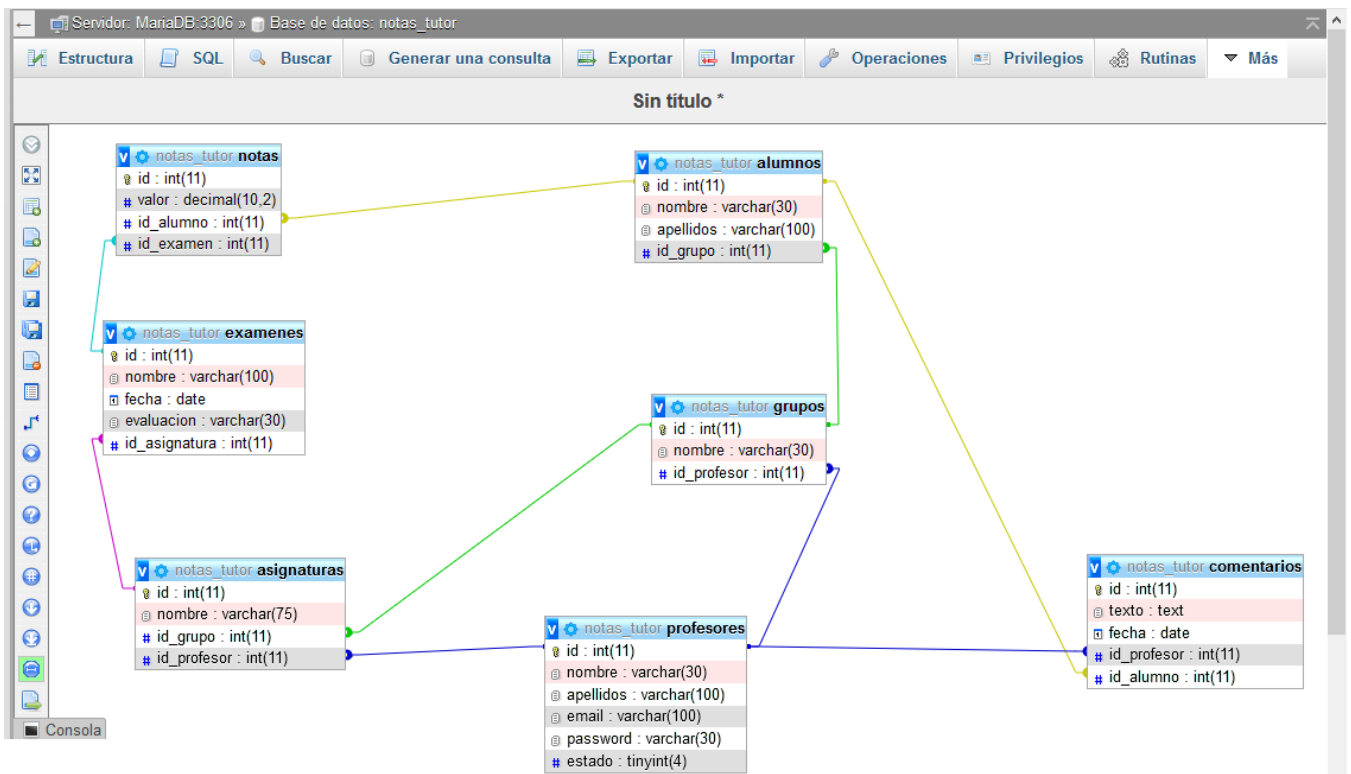


Ilustración 9 Vista gráfica de la base de datos con sus relaciones

El paso siguiente consiste en desarrollar una tabla en la que se disponen los contenidos que deberá tener la aplicación para dotarla de todas las funcionalidades necesarias para cumplir con los objetivos del producto que se han definido en el punto 1.3.2 de este documento. La tabla se ha dispuesto en niveles, de forma que se refleje la relación entre los contenidos y su ubicación dentro de las secciones de la aplicación, tal y como se propone en la Guía de la arquitectura de la información del Design Tool Kit de la UOC [6]

Nivel 1	Nivel 2	Nivel 3	Nivel 4
Inicio			
	Gestión Administrativa		
		Gestión de alumnos	
			Importar alumnos en bloque
			Crear/Modificar/Borrar /Consultar
		Gestión de profesores	
			Importar profesores en bloque
			Crear/Modificar/Borrar /Consultar
		Gestión de asignaturas	
			Importar asignaturas en bloque
			Crear/Modificar/Borrar /Consultar
		Gestión de clases	
			Importar clases en bloque
			Crear/Modificar/Borrar /Consultar
	Gestión de notas		
		Crear/Modificar/Borrar	
		Consultar	
			Por alumno
			Por asignatura
			Medias
	Ayuda		
		Tutorial	
		Solicitar ayuda	

Tabla 3 Tabla de contenidos estructurados por niveles

La tabla se ha hecho conforme a la información recogida hasta el momento en las fases anteriores y a nuestro criterio como diseñadores. Ahora, aplicando la metodología DCU debemos validar con el usuario si esta estructura es la correcta. Consultar con el usuario nos permitirá comprobar si falta o sobra algún contenido y si la distribución es la correcta y, además, conseguiremos que el resultado se base en sus propios modelos mentales, siéndole así más intuitivo. La técnica que utilizaremos para este propósito es un card sorting⁵, en este caso de método abierto, es decir, sin establecer categorías previas en las que los usuarios deban colocar las tarjetas.

⁵ La técnica conocida como Card Sorting consiste en hacer una tarjeta por cada uno de los contenidos de la tabla de inventario. Después se entregan las tarjetas al usuario para que las ordene según su criterio. Puede encontrarse información al respecto en Design Toolkit. Card Sorting [7]

3.2.2. Card Sorting

El card sorting se ha realizado con seis futuros usuarios, con diferentes perfiles basados en su experiencia docente y su habilidad como usuario informático. Las tarjetas que se les han facilitado a los usuarios son las siguientes:

Importar alumnos en bloque	Crear alumnos	Modificar alumnos	Borrar alumnos	Consultar alumnos
Importar profesores en bloque	Crear profesores	Modificar profesores	Borrar profesores	Consultar profesores
Importar asignaturas en bloque	Crear asignaturas	Modificar asignaturas	Borrar asignaturas	Consultar asignaturas
Importar clases/grupos en bloque	Crear clases/grupos	Modificar clases/grupos	Borrar clases/grupos	Consultar clases/grupos
Crear nota	Modificar nota	Borrar nota	Consultar notas por alumno	Consultar notas por asignatura

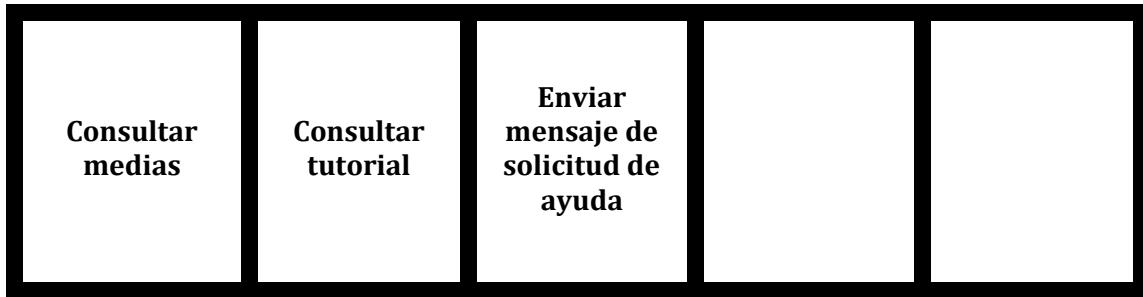


Tabla 4 Tarjetas para el Card Sorting. Modelo listo para imprimir y recortar

Se imprimen, se recortan y se barajan las tarjetas y se le entrega un juego a cada participante a fin de ver desde el punto de vista del usuario cómo están relacionados los contenidos entre sí y cómo unos están subordinados a otros. La prueba se realizó en el mismo centro, de manera individual y aprovechando patios u horas libres de los profesores

A continuación, se muestran los resultados obtenidos:

Usuario 1.

- Nivel técnico medio
- Experiencia docente entre 5 y 10 años
- Duración aproximada de la prueba 15 minutos



Ilustración 10 Card Sorting del usuario 1

Usuario 2

- Nivel técnico medio
- Experiencia docente menos de 3 años
- Duración aproximada de la prueba 15 minutos

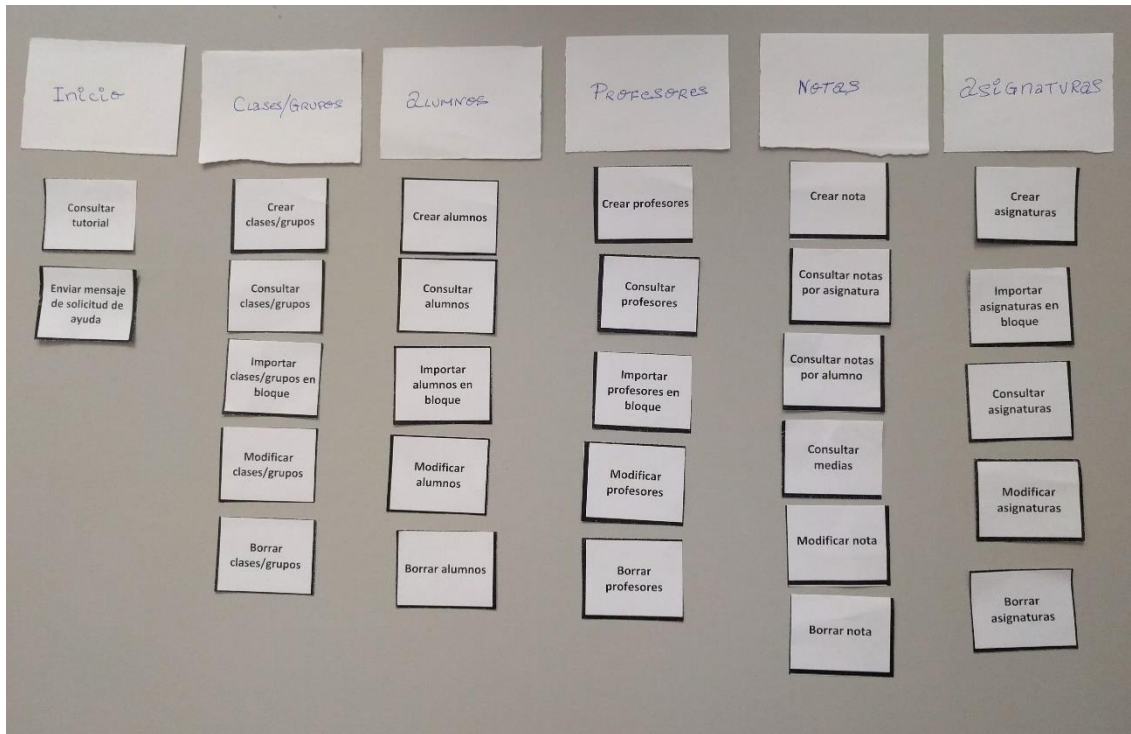


Ilustración 11 Card Sorting del usuario 2

Usuario 3

- Nivel técnico medio
- Experiencia docente entre 5 y 10 años
- Duración aproximada de la prueba 15 minutos

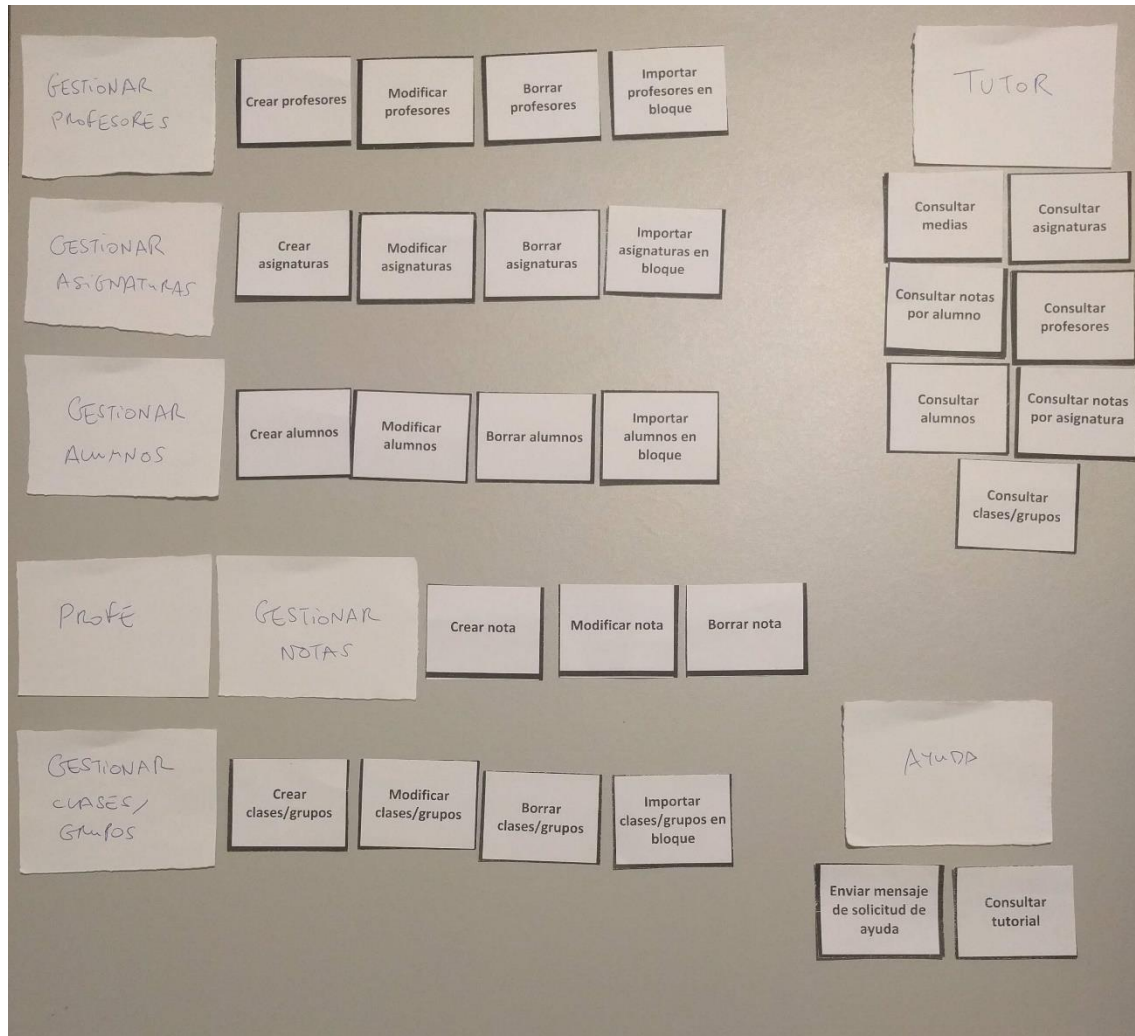


Ilustración 12 Card Sorting del usuario 3

Usuario 4

- Nivel técnico muy alto (es el otro informático del centro)
- Experiencia docente 5 años
- Duración aproximada de la prueba 15 minutos

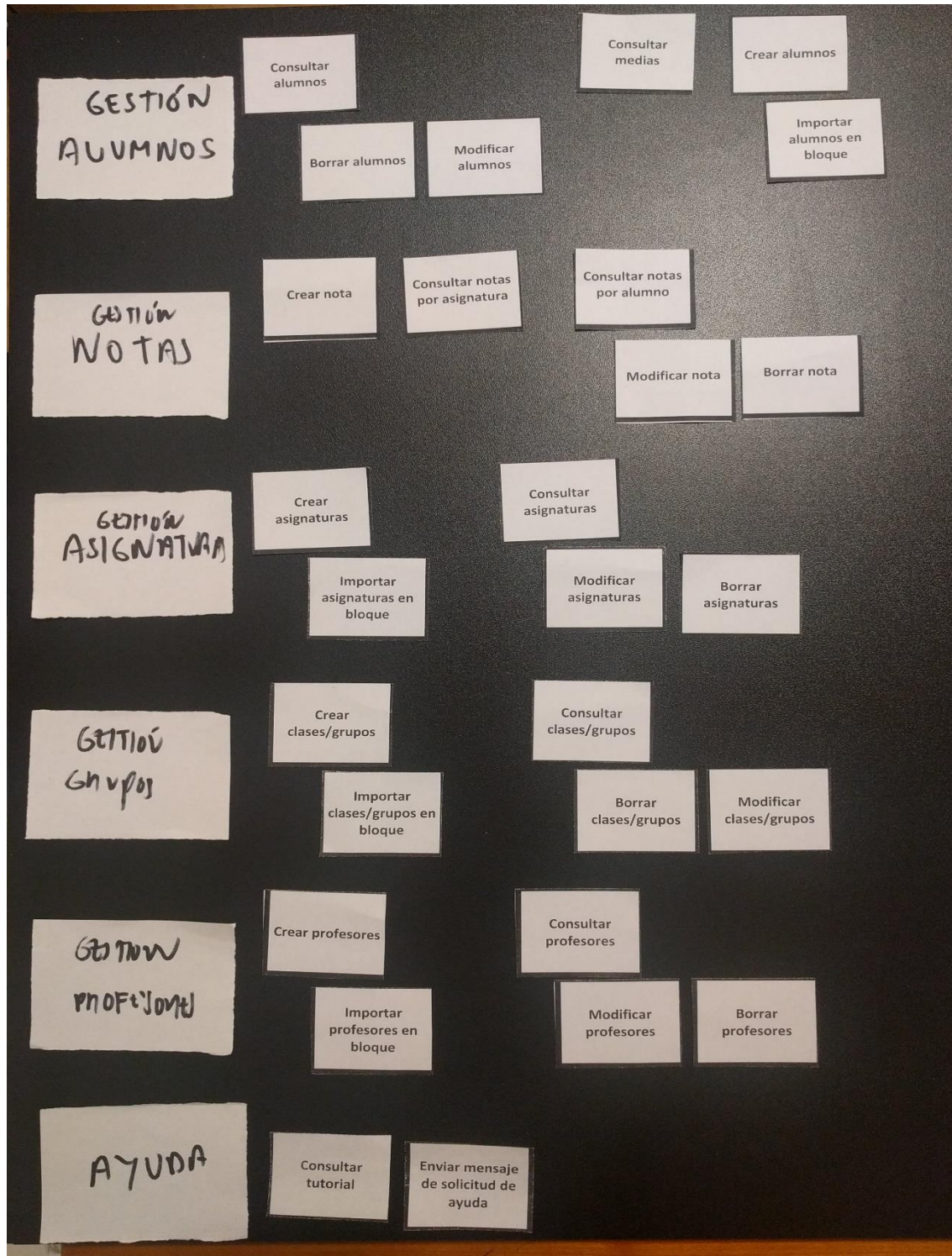


Ilustración 13 Card Sorting del usuario 4

Usuario 5

- Nivel técnico medio
- Experiencia docente menos de 1 años
- Duración aproximada de la prueba 15 minutos

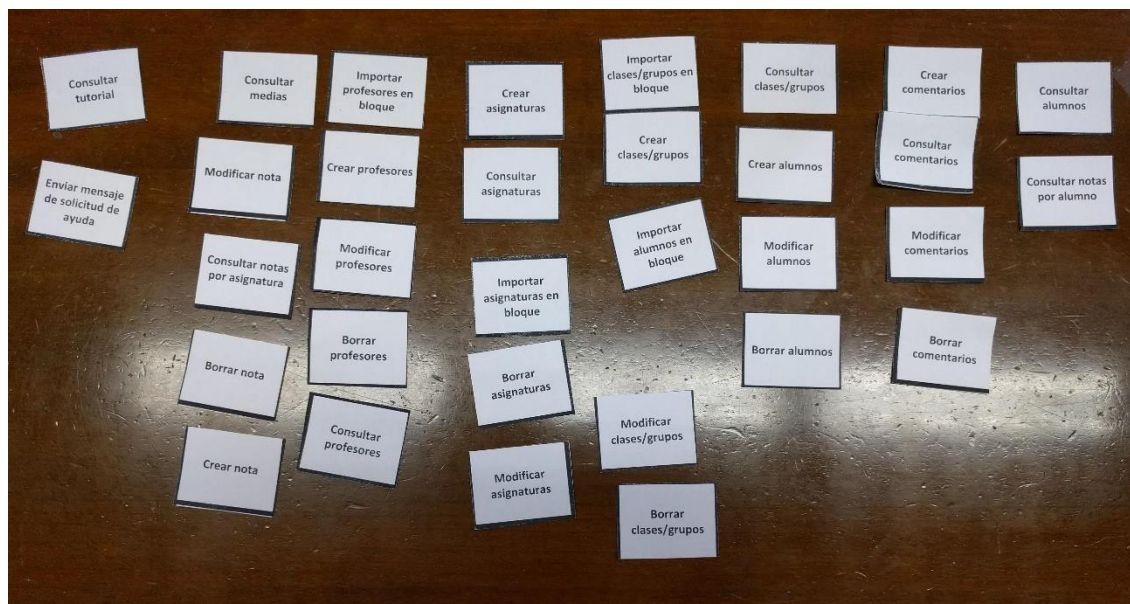


Ilustración 14 Card Sorting del usuario 5

Usuario 6

- Nivel técnico bajo/muy bajo
- Experiencia docente más de 25 años
- Duración aproximada de la prueba 15 minutos

Con la elección de este usuario se pretendía conseguir la aportación de alguien reactivo al uso de las tecnologías, pero el resultado no se consideró ni siquiera aprovechable. La actitud fue en todo momento de “yo no lo sé”, yo no lo entiendo”

Resultado del Card Sorting

Al observar los resultados obtenidos se concluye que:

- La mayor parte de los usuarios agrupan de forma muy similar los contenidos.
- Queda patente el hecho de que las tareas relacionadas con la creación de la base de datos (CRUD⁶ de profesores, cursos, alumnos, etc.) que normalmente se llevarán a cabo a principios de curso, provocan confusión en los usuarios. Es normal, ya que se trata de una tarea ajena a su trabajo y que será responsabilidad del informático del centro. Teniendo en cuenta esta circunstancia, se decide sacar del estudio toda la rama correspondiente a las tareas de administración, consistentes en general, en cargar la base de datos del instituto a principio de curso, y hacer alguna modificación puntual a lo largo del curso si, por ejemplo, llega un nuevo

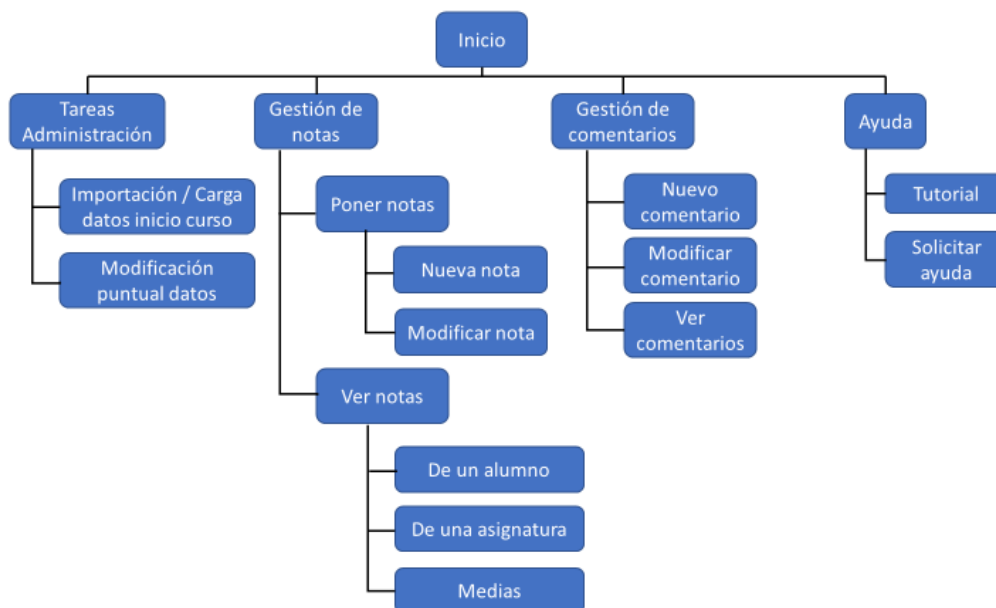
⁶ CRUD: Create, Read, Update, Delete que traduciríamos como Altas, Bajas Modificaciones y Consultas

alumno o cambia un profesor. En su lugar, en el menú se dejará simplemente una opción de “Petición al administrador” que facilitará el contacto con la persona encargada de realizar estas tareas de índole más técnica.

- Mejoras en el etiquetado

3.2.3. Árbol de contenidos (mapa del sitio)

Conforme con nuestro criterio inicial que sirvió de punto de partida, y añadiendo la información aportada por el card sorting, la estructura o árbol de contenidos que se decide implementar será la siguiente:



Finalmente, de acuerdo con el árbol de contenido propuesto, el menú principal de la aplicación quedaría:

- ✓ **Tareas de administración**
 - **Petición al administrador**
- ✓ **Poner notas**
 - **Nueva nota**
 - **Modificar nota**
- ✓ **Ver notas**
 - **De un alumno**
 - **De una asignatura**
 - **Medias por asignatura**
- ✓ **Poner comentarios**
 - **Nuevo comentario**
 - **Modificar comentario**
 - **Ver comentarios**
- ✓ **Ayuda**
 - **Tutorial**
 - **Solicitar ayuda**

3.2.4. Consideraciones personales

Cabe mencionar que el menú obtenido a partir del card sorting no difiere gran cosa del que se había planteado inicialmente en la tabla. De hecho, lo mismo ha sucedido después con los wireframes que se muestran en el punto siguiente: aportan menos cambios de los que yo esperaba. Además de por la sencillez del proyecto, esto puede ser debido o bien a mi falta de experiencia con estas técnicas, o bien al hecho de que, aunque actúe como diseñador, soy también profesor y tutor, lo que me ha permitido afinar bastante ya desde estos diseños iniciales. Sin embargo, incluso así, el DCU ha sido claramente provechoso (en algo en principio inesperado para mí) ya que ambas pruebas sí que han servido para poner de manifiesto que el "labeling" era mejorable. Concretamente, en la asignación inicial de nombre a los distintos ítems de menús y a las tarjetas del card sorting, probablemente debido a mi perfil informático, se utilizaron nombres que, aunque en "jerga informática" son más concretos y descriptivos, resultaron confusos para el usuario final, sobre todo cuando se trataba de usuarios poco habituados al uso de tecnologías de la información. El card sorting primero y las pruebas de usabilidad realizadas con los wireframes después, han puesto en evidencia esta circunstancia, ya que a menudo los usuarios solicitaban aclaraciones acerca de los términos empleados. Al observar este hecho, para mejorar el etiquetado (labeling) de la aplicación, siempre que el usuario pedía indicaciones sobre algún término, una vez explicado el significado se le solicitaba un término más descriptivo desde su punto de vista. Por ejemplo, "Crear nota" ha sido sustituido por "Poner nota", o en otra ocasión, en un cuadro de texto en el que se había empleado la etiqueta "Identificador de la nota" y se esperaban valores como "Examen tema 4", finalmente he puesto "Contenido evaluado", al parecer mucho más comprensible según el criterio de los usuarios entrevistados.

Otro aprendizaje de esta prueba ha sido la importancia de la elección de usuarios: un perfil de usuario informático demasiado bajo no es capaz de aportar información valiosa de lo que necesita. Sorprendentemente, me ha sucedido casi lo mismo con un usuario con conocimientos avanzados: en su afán de ayudar iba más allá, imaginaba pantallas y despletables, posiciones y tamaños de los elementos y no se centraba en los objetivos pedidos para esta prueba.

3.3. Rutas rojas.

Tal y como se explica en *The Fable of the User-Centered Designer* [18], en cualquier aplicación existen tareas prioritarias en las que debemos enfocarnos para que su funcionamiento sea óptimo, aún a costa de penalizar otras funcionalidades secundarias. Para nuestra aplicación, las rutas rojas son sin duda "**Poner notas**", "**Poner comentarios**" y "**Ver notas de un alumno**". A continuación, se definen unos escenarios y diagramas de flujo correspondientes a estas rutas rojas.

3.4. Escenarios.

3.4.1. Escenario 1: Poner una nueva nota

Alberto, profesor de Geografía e Historia del grupo de 2º B de la ESO, ha hecho un examen, del cual ya tiene los resultados y para que la tutora del grupo tenga fácil acceso a las notas ha decidido pasarlas a la nueva aplicación del centro. Para ello, desde el navegador del PC (Alberto se niega a utilizar su móvil personal para temas de trabajo) va a la página de la aplicación e introduce su usuario y contraseña. Una vez identificado, le aparece un menú en el que elige la opción “Poner notas” y dentro de esta, “Nueva nota”. Ya en la nueva pantalla introducirá la evaluación (1ª, 2ª o 3ª), el grupo y la asignatura, un texto identificador que le permitirá saber después a que prueba corresponden esas notas y la nota de cada alumno.

3.4.2. Escenario 2: Poner un comentario

Neus, profesora de Física y Química del grupo de 4º A de la ESO, ha pillado con chuletas llenas de fórmulas a P.T.S. Aparte de cómo afectará esto a la nota del examen, que ya se lo pensará, para que el tutor tenga constancia del hecho y pueda explicárselo a los padres del chico si lo considera conveniente, ha decidido anotararlo en la nueva aplicación del centro para estos menesteres. Para ello, nada más salir de clase, y con el navegador de su móvil, entra en la página de la aplicación y se identifica con su usuario y contraseña. Una vez identificado, en el menú que le aparece elige la opción “Poner comentarios”. En la página que se muestra elegirá grupo y asignatura, y después el alumno. Una vez hecho esto, introducirá la fecha y el comentario.

3.4.3. Escenario 3: Consultar notas de un alumno

Silvia, tutora del grupo de 4º de la ESO B, tiene previsto reunirse con los padres de su alumno J.R.R. el próximo jueves. Al concertar la cita ha recordado con alivio que dispone de información de las notas del alumno en la nueva aplicación “Ayuda al tutor”. Enciende su portátil, abre en el navegador la página de la aplicación y se identifica. Una vez dentro escoge en el menú la opción “Ver notas” y dentro de esta, “De un alumno”. En la nueva pantalla introduce el grupo y el alumno del cual necesita saber las notas y a continuación aparece un listado de todas las asignaturas con sus correspondientes notas.

3.5. Diagramas de flujo de interacción

Establecidas las rutas rojas y descritos los correspondientes escenarios, el siguiente paso será especificar los diagramas de flujo, que tal cual se puede leer en “Diagramas de flujo” del Design Toolkit de la UOC [8], “*permiten al diseñador sintetizar y analizar las diferentes interacciones del usuario con nuestra aplicación o servicio*”.

3.5.1. Diagrama de flujo escenario 1: Poner una nueva nota

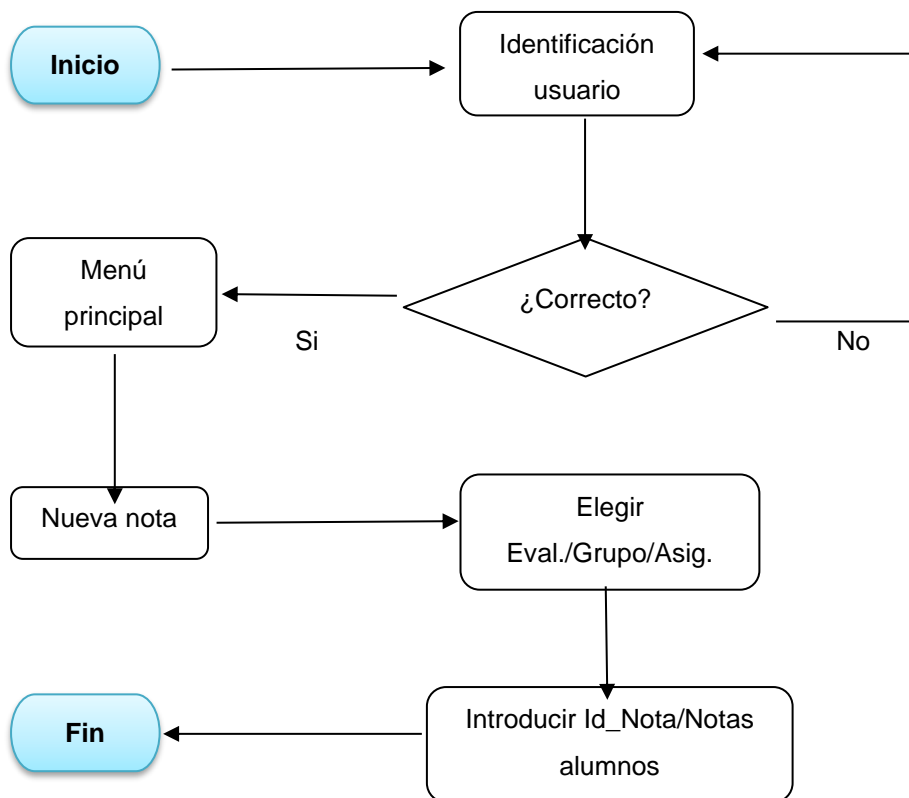


Ilustración 15 Diagrama de flujo del escenario 1

3.5.2. Diagrama de flujo escenario 2: Poner un comentario

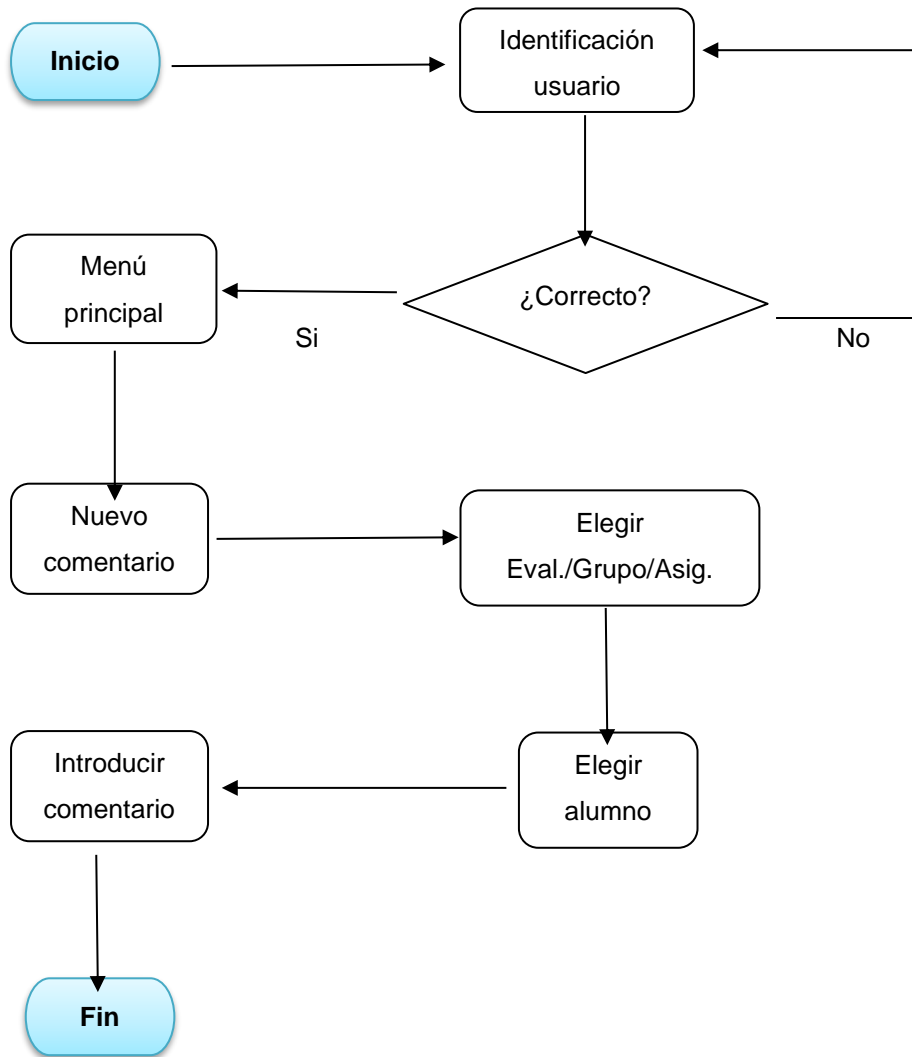


Ilustración 16: Diagrama de flujo del escenario 2

3.5.3. Diagrama de flujo escenario 3: Consultar notas de un alumno

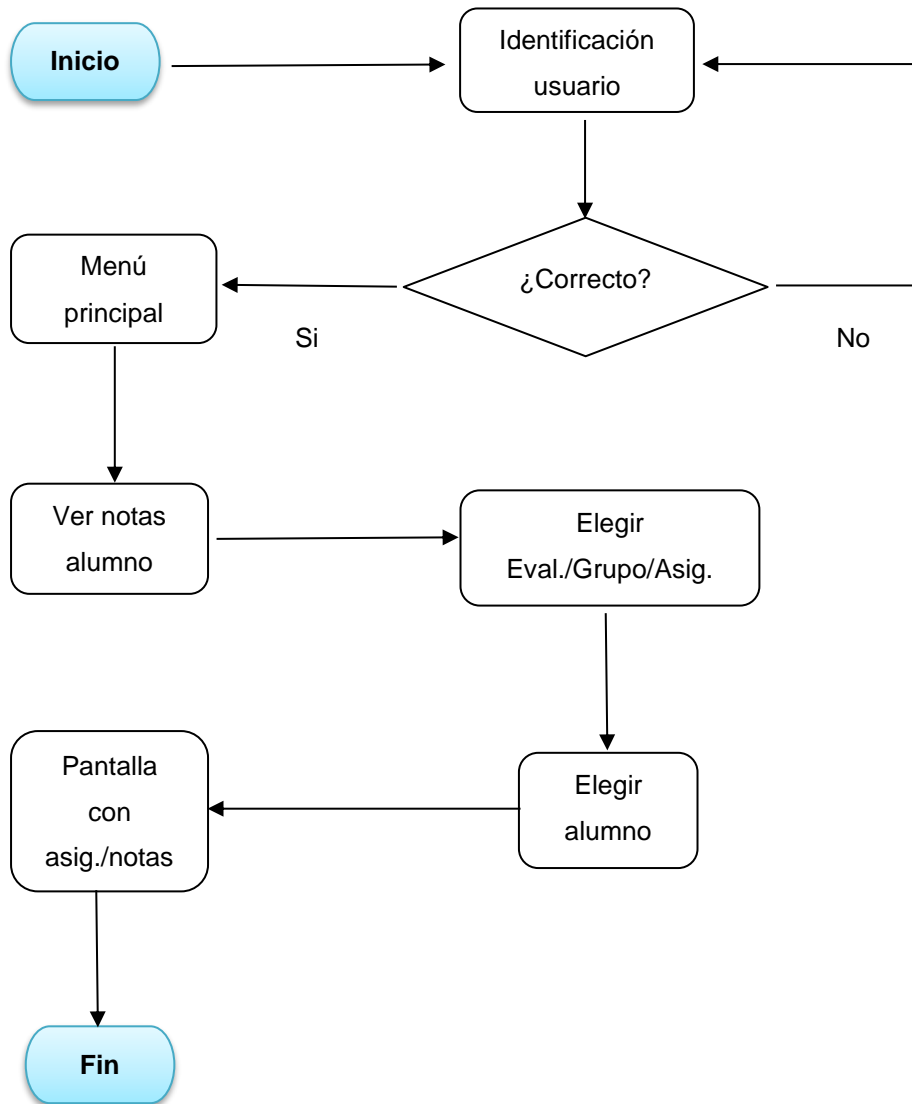


Ilustración 17 Diagrama de flujo del escenario 3

3.6. Prototipado

Esta fase pretende dotar de apariencia, hacer visibles, las funcionalidades que tendrá el producto para permitir después evaluarlo. En "Prototipado" dentro del Design Toolkit de la UOC [9] se propone hacer tres niveles de prototipos:

- Sketching
- Wireframes o prototipado de baja fidelidad
- Prototipado de alta fidelidad

En este caso, se han llevado a cabo solo los dos primeros. Para el tercero, en lugar de desarrollar un prototipo de alta fidelidad que permita al usuario cierta interacción, se ha implementado una primera versión de la aplicación real, que también ha provocado una nueva iteración a nivel de usabilidad antes de convertirse en el producto final que se presentado.

3.6.1. Sketching

La fase de Sketching consiste en realizar un primer prototipo de la aplicación que queremos desarrollar haciendo unos bocetos a mano de lo que serán las pantallas que verá el usuario. En este caso se utilizó una tableta y un lápiz óptico (stylus), pero igualmente valdría lápiz y papel.

Esta fase es creativa, en el sentido de que para obtener el resultado se aplica simplemente el criterio del diseñador. Además, este modelado rápido permite filtrar las ideas y descartar opciones inviables.

A modo de ejemplo a continuación se adjuntan algunos de los dibujos realizados:

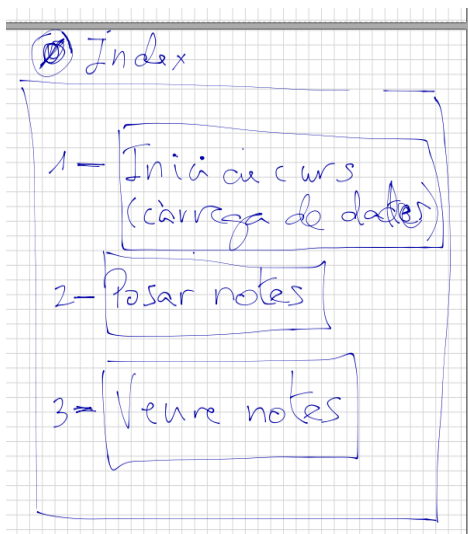


Ilustración 19 Sketching inicial del índice

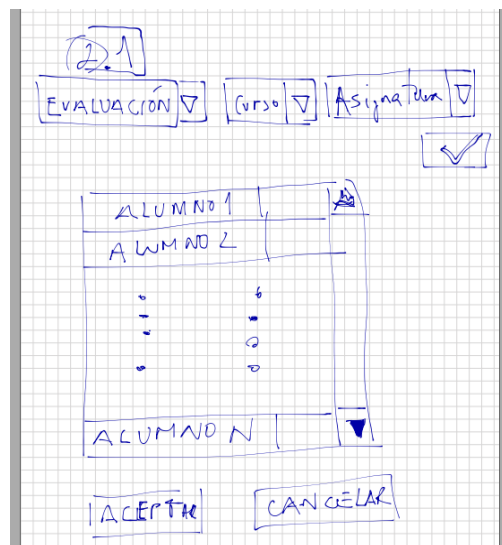


Ilustración 18 Sketching inicial de poner notas

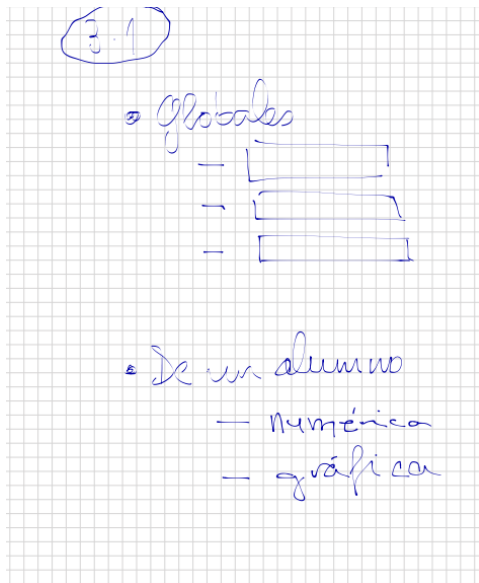


Ilustración 21 Sketching inicial de Ver notas

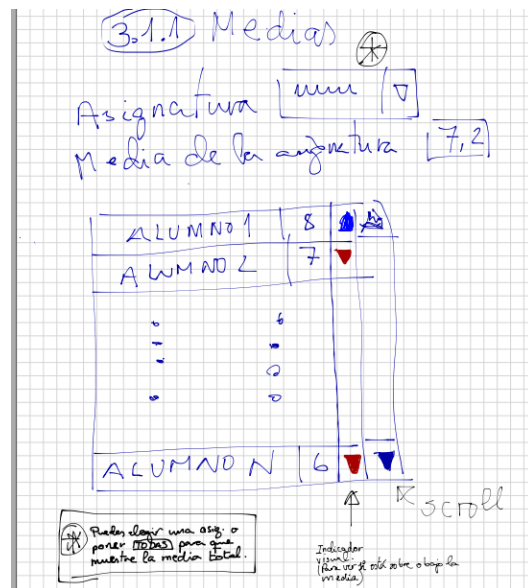


Ilustración 20 Sketching de Ver medias

Consideraciones personales

A destacar el hecho de que (al menos así ha sido en este caso) tan solo plasmar en papel la idea ya se producen las primeras iteraciones: se toma conciencia de enfoques inadecuados, se añaden opciones, se descartan otras y se intuye que otras quizás no serán realizables. Esta experiencia ha sido importante para valorar la potencia que tiene un entorno visual. El mejor resumen sería el dicho popular de que “una imagen vale más que mil palabras”.

3.6.2. Wireframes

Bien, en este punto del diseño ya tenemos una idea clara. Toca de nuevo mostrárselo al usuario para que dé su opinión, es decir, haga una primera evaluación y aporte un feedback que permita iterar de nuevo el diseño. Sin embargo, lo que tenemos hasta ahora es tan informal que puede no ser adecuado a este propósito. Debemos, por tanto, mejorar su apariencia, de modo que el usuario lo entienda y le sea posible entrar en situación. Este nuevo nivel de prototipo que presentaremos a los usuarios es lo que llamamos wireframes⁷.

El proceder que se ha seguido en la entrevista ha consistido, en primer lugar, en pedirle al usuario que piense en voz alta (“aloud thinking”) y que tenga siempre presente que no se le evalúa a él, sino a la interfaz, tal y como propone la página web “Como realizar un test de usabilidad en cinco pasos” [3]. Después se le ha dado una pequeña explicación de los objetivos de la aplicación teniendo los wireframes delante, y finalmente, sometiéndole a observación para ver cómo se desenvuelve, se le ha solicitado:

⁷ En este TFG la herramienta utilizada para dibujar los wireframes ha sido Balsamiq Wireframes para Google Drive. Los motivos de la elección han sido su facilidad de uso y aprendizaje y el precio, tan solo 5\$ mensuales.

- Si fueras un profesor (no tutor) y quisieras poner una nota, ¿cómo lo harías?
- Si fueras tutor de un grupo y quisieras consultar las notas de un alumno, ¿cómo lo harías?
- ¿Hay algo importante que pienses que falta?
- Ya para finalizar, aprovechando la cercanía con los usuarios por parte del entrevistador, dado que se trata de compañeros de trabajo, se han mantenido unos minutos de charla acerca del tema.

Siguiendo con el criterio de centrar los esfuerzos en nuestras rutas rojas, sobre los wireframes correspondientes a esas rutas rojas es sobre los que se ha hecho mayor incidencia al presentárselos a los usuarios.

A continuación, se muestra el prototipo presentado al usuario:



Ilustración 23 Primeros wireframes. Pantalla Inicio

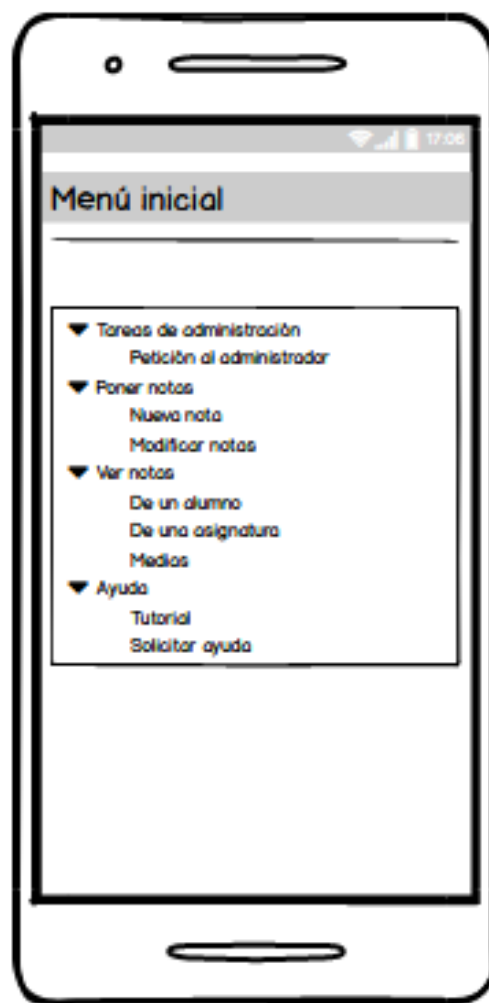


Ilustración 22 Primeros wireframes. Menú principal

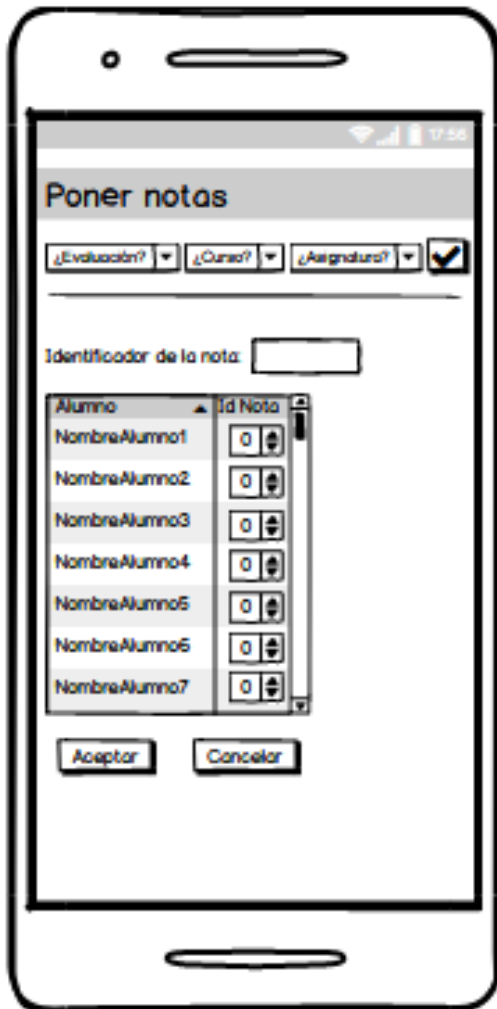


Ilustración 25 Primeros wireframes. Poner notas

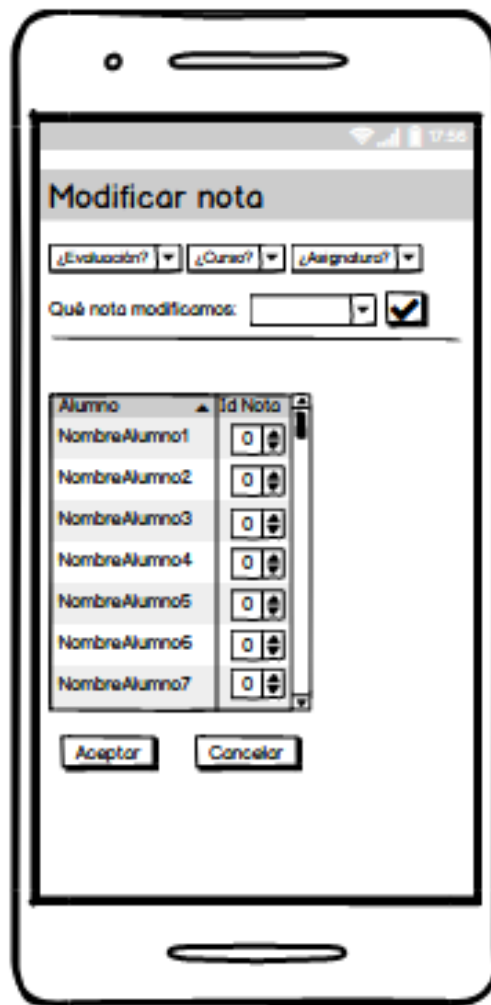


Ilustración 24 Primeros wireframes. Modificar notas

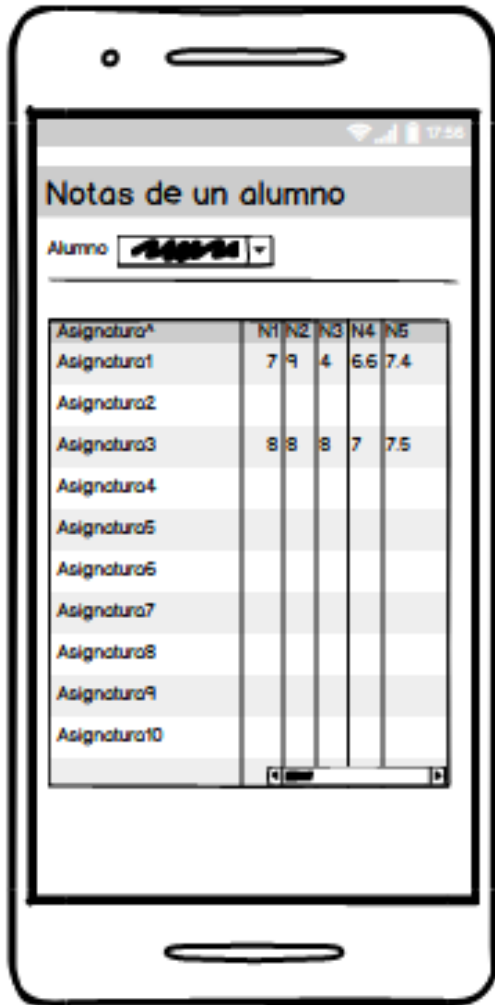


Ilustración 27 Primeros wireframes. Notas de un alumno

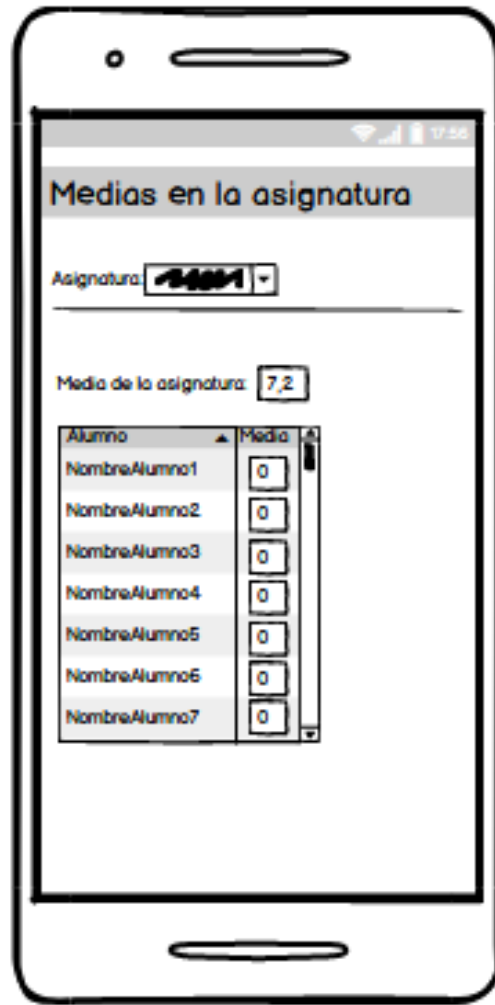


Ilustración 26 Primeros wireframes. Ver medias

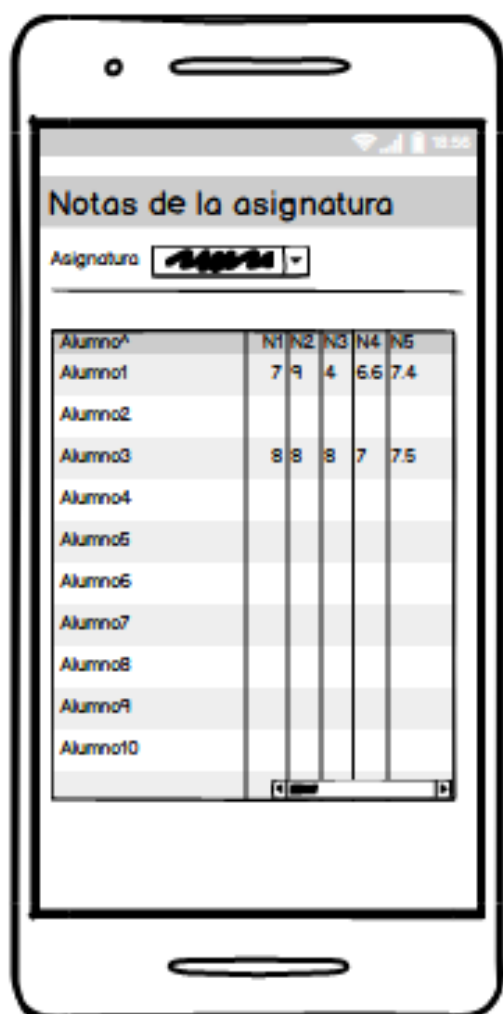


Ilustración 28 Primeros wireframes. Notas de una asignatura

3.6.3. Pruebas de usabilidad realizadas

Las pruebas se realizaron de manera individual, en la sala de profesores, en una hora libre en que coincidieran entrevistado y entrevistador. La duración no excedió nunca de 20 minutos incluyendo la charla informal posterior. Pese a haber visto en otros casos que en la ficha del entrevistado es habitual especificar sexo, profesión y nivel de estudios, en nuestro caso se ha obviado, ya que el primero es irrelevante en este caso y los otros dos son siempre “Profesor/a de secundaria” y “Estudios universitarios”.

Usuario 1

Nivel técnico muy alto

Edad 32 años

Tiempo empleado inferior a 5 minutos

- **Si fueras un profesor (no tutor) y quisieras poner una nota, ¿cómo lo harías?**
Ha elegido el menú correcto y habría puesto la nota sin problemas ni errores

- **Si fueras tutor de un grupo y quisieras consultar las notas de un alumno, ¿cómo lo harías?**
Ha elegido correctamente la opción en el menú y luego el desplegable del alumno.
- **¿Hay algo importante que pienses que falta?**
Contraseña en pantalla de identificación

Usuario 2

Nivel técnico medio-bajo

Edad 59 años

Tiempo empleado 9 minutos

- **Si fueras un profesor (no tutor) y quisieras poner una nota, ¿cómo lo harías?**
Ha elegido correctamente la opción del menú, pero no entiende los menús desplegables de elección de evaluación, grupo y asignatura. Habría ido directamente a poner notas sin informar esos campos. Sin explicaciones adicionales no habría sido capaz de poner las notas.
Pregunta que hay que poner en el campo "Identificador de la nota"
- **Si fueras tutor de un grupo y quisieras consultar las notas de un alumno, ¿cómo lo harías?**
Ha elegido bien la opción del menú y ahora sí, el desplegable del alumno
- **¿Hay algo importante que pienses que falta?**
Además de notas, poder poner comentarios sería muy útil para el tutor

Usuario 3

Nivel técnico medio-alto

Edad 45 años

Tiempo empleado 5 minutos

- **Si fueras un profesor (no tutor) y quisieras poner una nota, ¿cómo lo harías?**
- Ha elegido el menú correcto y habría puesto la nota sin problemas ni errores
- **Si fueras tutor de un grupo y quisieras consultar las notas de un alumno, ¿cómo lo harías?**
Ha elegido bien la opción del menú y habría clicado sobre el desplegable del alumno correctamente
- **¿Hay algo importante que pienses que falta?**
Que las notas tengan dos decimales

Usuario 4

Nivel técnico medio

Edad 39 años

Tiempo empleado 7 minutos

- **Si fueras un profesor (no tutor) y quisieras poner una nota, ¿cómo lo harías?**
- Ha elegido el menú correcto. Una vez en la pantalla no veía claro el uso de los desplegables
- **Si fueras tutor de un grupo y quisieras consultar las notas de un alumno, ¿cómo lo harías?**

Ha elegido bien la opción del menú y habría clicado sobre el desplegable del alumno correctamente

- **¿Hay algo importante que pienses que falta?**
Datos de contacto de los padres (teléfono). Añadir comentarios además de notas

Usuario 5

Nivel técnico medio-bajo

Edad 56 años

Tiempo empleado 11 minutos

- **Si fueras un profesor (no tutor) y quisieras poner una nota, ¿cómo lo harías?**
Ha elegido el menú correcto y tras dudar, finalmente decide correctamente como sería el proceso de poner la nota. Pregunta que hay que poner en el campo "Identificador de la nota"
- **Si fueras tutor de un grupo y quisieras consultar las notas de un alumno, ¿cómo lo harías?**
Ha entrado en las tres opciones que hay para consultas. Finalmente opta por la correcta
- **¿Hay algo importante que pienses que falta?**
Comentarios sobre los alumnos

3.6.4. Nuevos wireframes después de las pruebas

El prototipo ha sido bien aceptado y no se han sugerido grandes variaciones, con lo que, después de las pruebas de usabilidad, los cambios que se han aplicado finalmente para la siguiente iteración son:

- Se ha mejorado el labeling: al igual que lo que ya sucedió con el card sorting, una pregunta tipo ¿esto qué me habías dicho que quería decir? ha sido un indicador claro y recurrente de que un término necesitaba ser revisado.
- Se ha añadido la funcionalidad de inclusión de comentarios: inicialmente la aplicación solo contemplaba el tratamiento de las notas. En las pruebas de usabilidad, igual que sucedió en alguna entrevista inicial, se manifestó por parte de los usuarios que incluir comentarios del tipo "habla mucho en clase", "no acepta bien que se le llame la atención" sería también muy útil. A priori, técnicamente no parece difícil de implementar (básicamente repite, pero más simplificado, el mismo proceso que poner / ver una nota) y es cierto que aporta una funcionalidad de alto valor añadido. Así pues, no tan solo se ha añadido la funcionalidad, sino que además se la ha considerado como una nueva ruta roja.
- Se han detectado pequeños fallos u olvidos. Por ejemplo, faltaba el campo contraseña en la pantalla de inicio.

Tras aplicar estas propuestas, finalmente los nuevos wireframes quedarán así:



Ilustración 30 Wireframes mejorados. Pantalla de inicio

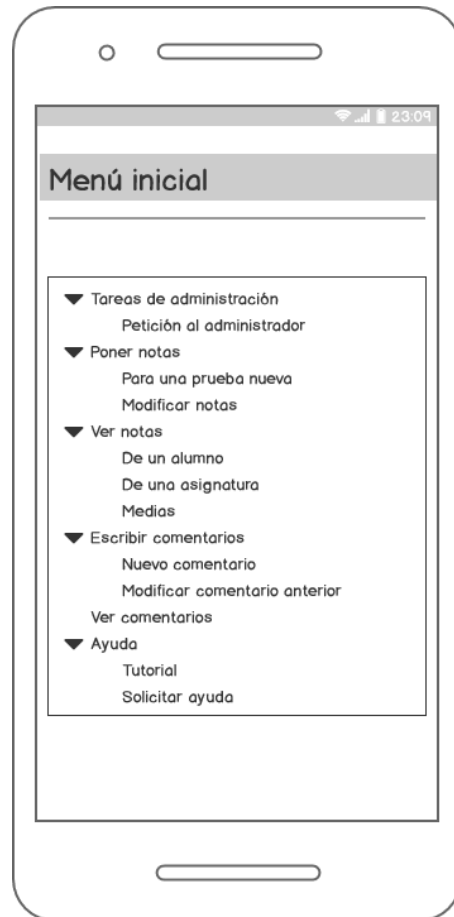


Ilustración 29 Wireframes mejorados. Pantalla de menú principal.

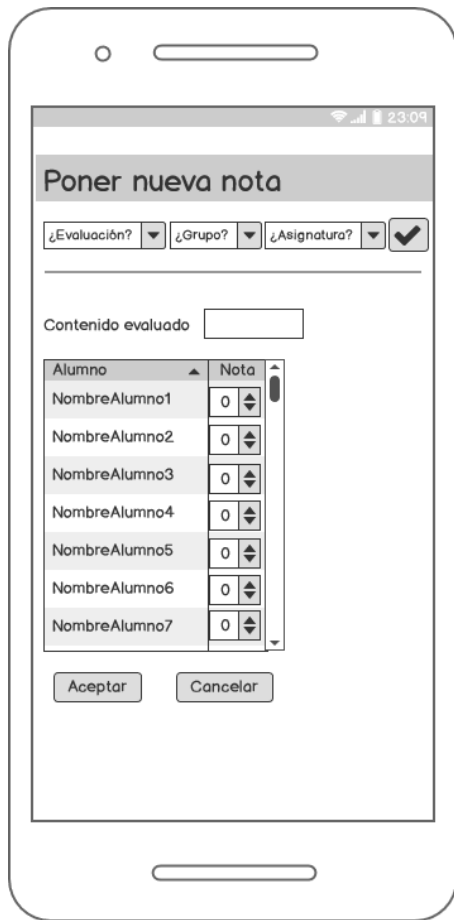


Ilustración 32 Wireframes mejorados. Poner nueva nota

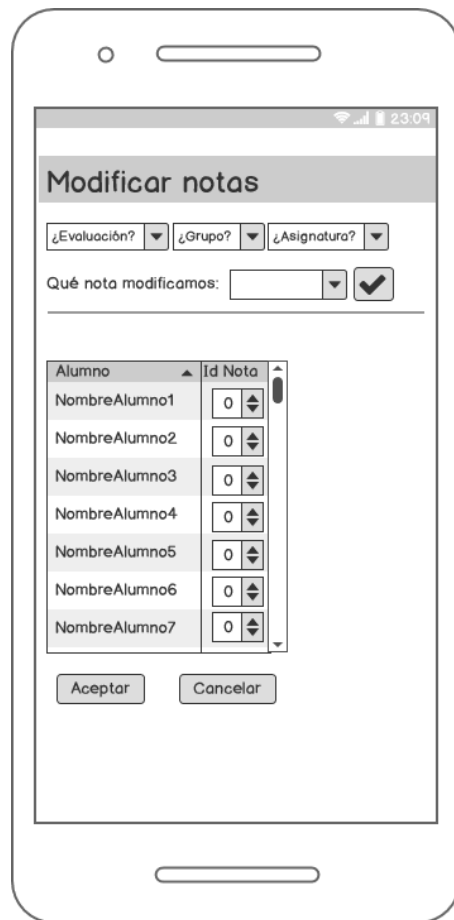


Ilustración 31 Wireframes mejorados. Modificar nota

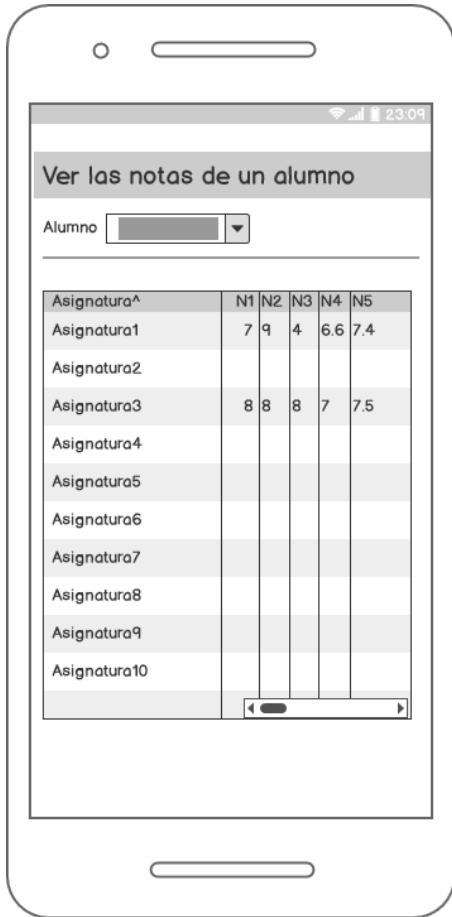


Ilustración 34 Wireframes mejorados. Notas de un alumno

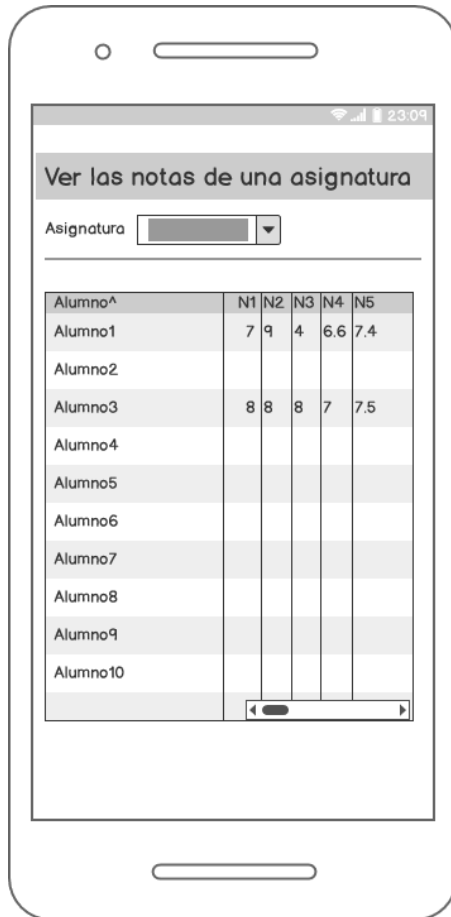


Ilustración 33 Wireframes mejorados. Ver notas de una asignatura

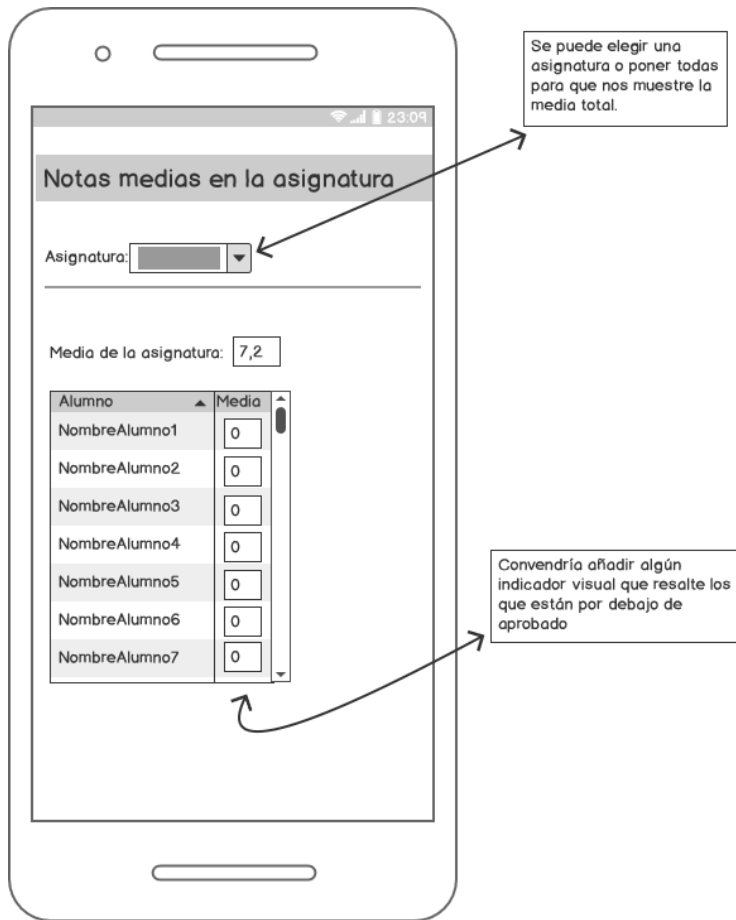


Ilustración 35 Wireframes mejorados. Medias de asignatura

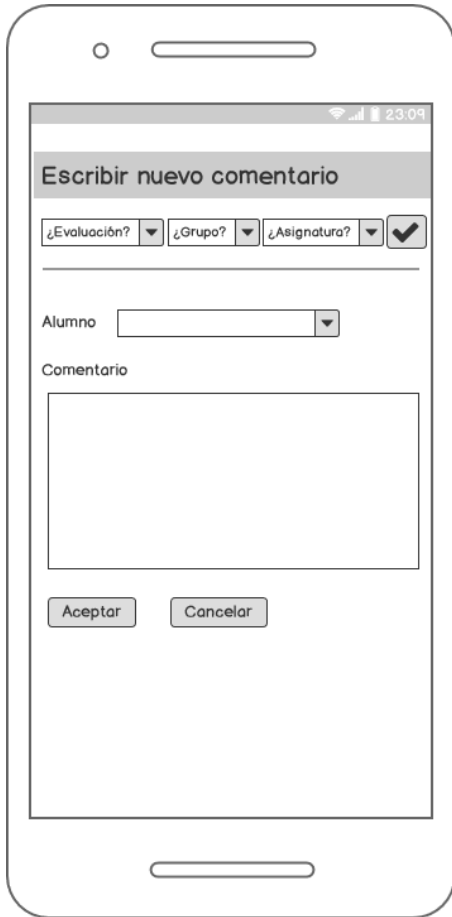


Ilustración 37 Wireframes mejorados. Poner comentario

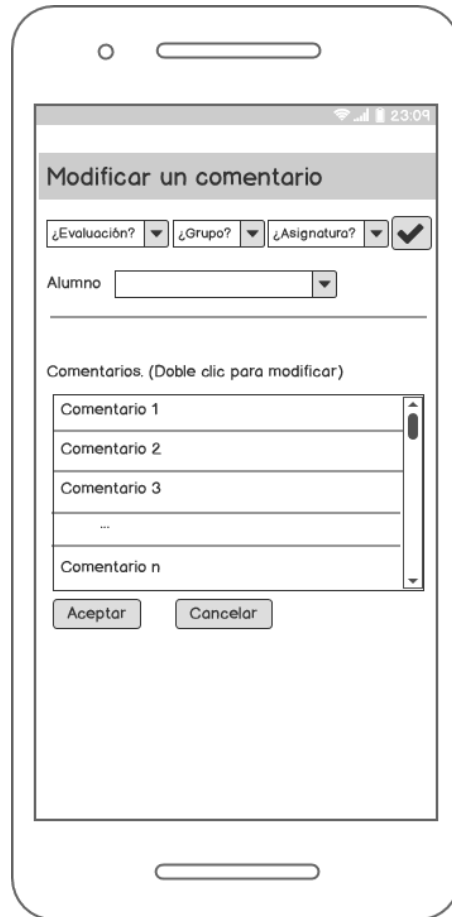


Ilustración 36 Wireframes mejorados. Modificar comentario

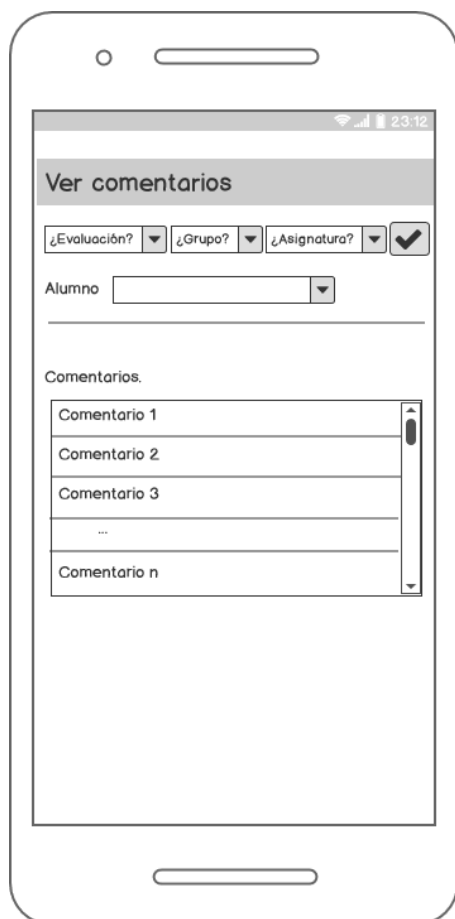


Ilustración 38 Wireframes mejorados. Ver comentarios

3.6.5. Siguiendo paso

Llegados a este punto, se dispone de un diseño con una arquitectura y unas funcionalidades que se ha comprobado que satisfacen las necesidades de los futuros usuarios de la aplicación. Surgen ahora dos posibilidades:

- Realizar un nuevo prototipo de alta fidelidad. Este tipo de prototipos simulan la aplicación final con la máxima exactitud posible, intentando respetar colores, tipos de letras, controles, ... Además, suelen simular también comportamiento, aceptando entradas de ratón, teclado o pantalla en determinadas zonas, y respondiendo del mismo modo en que lo haría la interfaz real en lo que atañe al flujo de pantallas, mensajes, cambios de estado...
- Implementar, con programación real, la primera versión del producto

Tanto si se elige hacer un prototipo de alta fidelidad, como si se pasa ya a programar la aplicación, el resultado obtenido puede y debe ser sometido de nuevo a pruebas de usabilidad con el usuario y, probablemente, a una nueva iteración aplicando las mejoras que surjan.

Normalmente, por economía, las empresas llevan a cabo la siguiente iteración realizando un prototipo de alta fidelidad. Sin embargo, dado que para este TFG las motivaciones son el aprendizaje y la adquisición de una experiencia global, viviendo el desarrollo de un proyecto desde la concepción de la

idea hasta el lanzamiento de la aplicación, se ha considerado más interesante optar por la segunda opción, la implementación de la aplicación real.

3.6.6. Estudio del prototipo real implementado

Los prototipos que se han desarrollado tenían un objetivo: ofrecer al usuario una representación gráfica que pudiera entender, para pedirle su opinión y asegurarnos de que la funcionalidad es completa y la estructura adecuada. Esto lo hemos hecho mediante las pruebas de usabilidad.

Sin embargo, al llevar a cabo la implementación ha surgido un problema: el aspecto que tenían los wireframes no es exactamente reproducible, o bien, aun siéndolo, la realidad demuestra que no es práctico. Por ejemplo, recordemos el wireframe de la pantalla de inicio de la aplicación

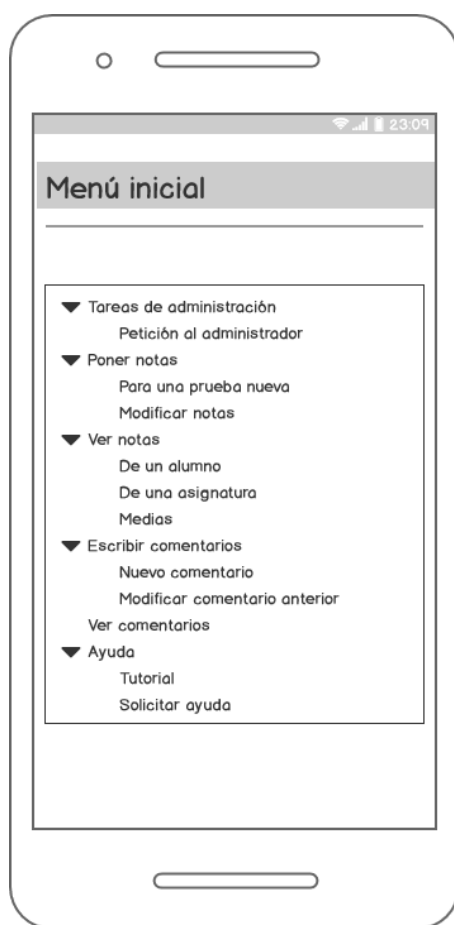


Ilustración 39 Ejemplo de menú no aconsejado para su implementación

En una pantalla de móvil tanta letra ha resultado verse muy pequeña, además, con las opciones tan juntas al tocar con el dedo es fácil pulsar sobre una no deseada.

Ante esta circunstancia, el prototipo real implementado ha cambiado su aspecto. Sabemos que respeta la funcionalidad, pero podría haber perdido en usabilidad. De acuerdo con la filosofía subyacente al DCU debemos consultar de nuevo al usuario y según el resultado de esta consulta, iterar el prototipo.

NOTA 1: PARA QUE LA APLICACIÓN SEA OPERATIVA SE HA PROCEDIDO A RELLENAR CON DATOS FICTICIOS LAS TABLAS. NO SE HA VULNERADO EN NINGÚN MOMENTO LA LEY DE PROTECCIÓN DE DATOS

Para las siguientes pruebas de usabilidad, se ha decidido enviar a un grupo de profesores conocidos el siguiente email:

Estimado/a xxxxxxxxxx

Como ya sabes estoy acabando mi TFG de Ingeniería Informática. Necesito que me ayudes a testear la aplicación que he desarrollado. Para ello, por favor visita la dirección web

<http://rafaelsoriano-uoc-tfg.com> y entra con los siguientes datos:

Usuario: xxxxxxxxxx

Contraseña: xxxxxxxxxx

La aplicación básicamente permite a los profesores poner notas y a los tutores verlas, de forma que, en caso de visita de un padre de alumno, el tutor disponga de esos datos de manera inmediata.

Tu tarea:

1. Pon nota de una prueba a toda la clase de 1º ESO A
 - a. ¿Has podido hacerlo?
 - b. ¿Qué has utilizado? Tablet – Móvil – PC
 - c. Ha sido: fácil – normal – difícil
2. Si has podido hacerlo, repite el proceso dos veces y pon dos notas más a toda la clase de 1º ESO A
3. Poner 3 comentarios (a tres alumnos diferentes, a poder ser de la misma clase anterior)
 - a. ¿Has podido hacerlo?
 - b. Fácil – normal – difícil
4. Desconéctate y vuelve a entrar, ahora con el usuario: sribes@micolegio.es // 123456
 - a. ¿Has podido hacerlo?
 - b. Fácil – normal – difícil
5. Con este usuario ahora eres tutor/a del grupo de 1º ESO A. ¿Qué notas tiene en la primera evaluación la alumna Alejandra Lucero Alonso?
 - a. ¿Has podido hacerlo?
 - b. Fácil – normal – difícil
6. ¿Tiene puesto algún comentario?
 - a. ¿Has podido hacerlo?
 - b. Fácil – normal – difícil
7. Para terminar, ¿consideras la aplicación fácil? Sí / No
8. ¿Los tamaños de letra son adecuados? Sí / No
9. ¿Los diferentes botones son fáciles de manejar? Sí / No
10. ¿Cuánto tiempo te ha llevado realizar esta prueba?


¡¡Gracias por tu ayuda!!


Los usuarios que se les han enviado aparecen todos como profesores del grupo 1º ESO A y son los siguientes:

Usuario	Contraseña
aarreola@micolegio.es	123456
padame@micolegio.es	123456
lmarin@micolegio.es	123456
lcepeda@micolegio.es	123456
aesquivel@micolegio.es	123456
acarrera@micolegio.es	123456
aguasch@micolegio.es	123456
raranda@micolegio.es	123456
nestevez@micolegio.es	123456
jgastelu@micolegio.es	123456
sribes@micolegio.es (tutor de 1º ESO A)	123456

Tabla 5 Listado de usuarios facilitados para pruebas de usabilidad con la aplicación real

Estas tareas son las rutas rojas de la aplicación, por eso se ha planteado sobre ellas la prueba de usuarios. Las pantallas que han tenido que ver los usuarios para realizar la prueba son las que se muestran a continuación. Si se desea ver el resto de la aplicación y probarla, se puede acceder con cualquiera de los usuarios anteriores, o de los que aparecen en la lista completa que se adjunta en el Anexo 3


Ayuda al Tutor





Identificación


Ilustración 40 Pantalla de login


Ayuda al Tutor Hola, Santiago [Desconectar](#)


Profesores



 Poner Notas


 Ver Notas


 Modificar Nota


 Poner Comentario


 Ver Comentarios


 Modificar Comentario

Tutores



 Seguimiento del alumno

Ilustración 41 Pantalla de inicio. Menú principal de la aplicación.

Ayuda al Tutor Hola, Araceli [Desconectar](#)

Portada / Poner Nota

Poner Nota

1ª Evaluación ▾

1º ESO A ▾

Matemàtiques ▾

Mostrar Alumnos

Contenido evaluado

Descripción

Lista de alumnos

Apellidos	Nombre	Nota
Anaya Ávila	Natalia	<input style="width: 100%;" type="text"/>
Arteaga Miranda	Sara	<input style="width: 100%;" type="text"/>
Ruelas Sánchez	Hidaya	<input style="width: 100%;" type="text"/>
Sarabia Sandoval	Inés	<input style="width: 100%;" type="text"/>
Tirado Hurtado	Manel	<input style="width: 100%;" type="text"/>
Vanegas Salas	Daniela	<input style="width: 100%;" type="text"/>

[Guardar notas](#)

Ilustración 42 Pantalla Poner notas

[Portada](#) / [Poner Comentario](#)

Poner Comentario

1ª Evaluación ▾ 1º ESO A ▾ Matemàtiques ▾ Mostrar Alumnos

Alumno

Alumno ▾

Comentario

Guardar comentario

Ilustración 43 Pantalla Poner comentarios

Seguimiento del Alumno

1º ESO A

Lucero Alonso, Alejandra

Mostrar Datos

Primera Evaluación

Segunda Evaluación

Tercera Evaluación

Notas

Biología i Geologia

Nota Media: 4.50



Educació física

Nota Media: -



Educació plàstica visual i audiovisual I

Nota Media: 7.40



Geografia i història

Nota Media: 8.00



Llengua castellana i literatura

Nota Media: 6.50



Llengua catalana i literatura

Nota Media: -



Matemàtiques

Nota Media: 5.00



Música

Nota Media: 4.00



Primera llengua estrangera: anglès

Nota Media: -



Religió

Nota Media: -



Comentarios

Anna Arreola - 7/1/2020 16:27:36h

Alejandra no está muy centrada esta primera evaluación

3.6.7. Resultados de las pruebas de usabilidad del prototipo real implementado

Las pruebas de usabilidad se han realizado con 10 usuarios con un perfil muy homogéneo, todos ellos son profesores de secundaria y por tanto titulados universitarios. A fin de definir un poco más el perfil, se adjunta también su experiencia profesional y su nivel como usuario informático.

Usuario	Experiencia	Nivel informático
Usuario 1 (Anna)	Más de 15 años	Muy bueno
Usuario 2 (Pepe)	Más de 15 años	Bueno
Usuario 3 (Laura)	Menos de 5 años	Bueno
Usuario 4 (JM)	Más de 15 años	Bueno
Usuario 5 (Toñi)	Más de 15 años	Medio
Usuario 6 (Nerea)	Menos de 5 años	Bueno
Usuario 7 (Araceli)	Más de 15 años	Medio/Bajo
Usuario 8 (Rocío)	Entre 5 y 15 años	Bueno
Usuario 9 (Silvia)	Entre 5 y 15 años	Medio/Bueno
Usuario 10 (Santi)	Más de 15 años	Medio/Bajo

Interesa saber:

- Dispositivo empleado para conectarse (PC, móvil o tableta)
- Si son capaces de realizar las tareas principales que hemos definido como rutas rojas
- Nivel de dificultad de cada tarea
- Percepción general del nivel de dificultad
- Detectar si textos y botones son de tamaño adecuado para trabajar o demasiado pequeños (aplica sobre todo para móviles)
- Tiempo aproximado empleado (se estima que debería estar entre 10 y 15 minutos)

Se adjunta en la página siguiente una tabla resumen con las contestaciones de los usuarios

Tarea	Usuario 1	Usuario 2	Usuario 3	Usuario 4	Usuario 5	Usuario 6	Usuario 7	Usuario 8	Usuario 9	Usuario 10
Dispositivo	PC	Tablet	Móvil	Móvil	PC	Móvil	PC	Móvil	Móvil	Tablet
Poner nota	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Fácil/Normal/Difícil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Normal	Fácil	Fácil
Repetición	Sí*	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Poner comentario	Sí*	Sí	Sí	Sí	Sí	Sí*	Sí	Sí	Si*	Sí
Fácil/Normal/Difícil	Fácil	Normal	Fácil	Fácil	Fácil	Normal	Fácil	Fácil	Normal	Fácil
Logout/Login	Sí	Si	Sí	Sí	Sí	Si	Sí	Si	Si	Si
Fácil/Normal/Difícil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil
Consultar notas (tutor)	Sí	Sí	Sí	Sí	Sí	Sí *	Sí	Sí	Sí *	Sí
Fácil/Normal/Difícil	Fácil	Fácil	Fácil	Normal*	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil
Consultar comentarios (tutor)	Sí	Sí	Sí	Sí	Sí *	Sí *	Sí	Sí	Sí *	Sí
Fácil/Normal/Difícil	Fácil	Fácil	Fácil	Fácil	Fácil	Fácil	Normal	Normal	Fácil	Fácil
¿Aplicación fácil?	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Legible	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Manejo adecuado	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí	Sí
Tiempo empleado	10 min	15 min	15 min	10 min	15-20 min	15 min	15-20 min	10 min	10 min	15 min

Tabla 6 Resumen de la información recogida en las pruebas de usabilidad

Las contestaciones marcadas con * indican que aparte de lo que indica el resultado ha habido alguna incidencia o comentario

El 70% de los usuarios han elegido móvil o Tablet, lo cual demuestra que optar por web responsive fue un acierto.

Los usuarios califican la aplicación como de uso fácil y en comentarios aparte dos profesores coinciden en los adjetivos clara y limpia.

En cuanto a las mejoras sugeridas y fallos detectados encontramos:

Usuarios 1: He tenido que ir a "Modificar nota" porque al poner notas he puesto algunas con coma en lugar de punto y no se han guardado bien ¿Sería posible que el campo de texto donde se introduce la nota ya controlara que sólo se están metiendo números? (Usuario 3 y Usuario 4 comentan lo mismo)

Usuario 1: Después de darle a "Guardar comentario", el campo de texto permite seguir introduciendo datos, aunque al final no los guarda, ya que que el botón "Guardar comentarios" sí que está desactivado. Esto genera confusión. (Usuario 6 y usuario 9 algo parecido)

Usuario 5: Cuando el tutor mira las notas de un alumno, para ver el detalle hay que darle a la punta de flecha. Casi no se ve. Se podría remarcar más (Usuario 6, con móvil, coincide).

Usuario 8: Al poner notas es un poco rollo tener que ir cambiando el teclado del móvil de texto a numérico.

Además, se ha detectado el siguiente problema, principalmente con los usuarios de móvil: al entrar como tutor e ir a ver las notas, en lugar de elegir la opción del tutor, se metían en "Ver notas" como profesor, con lo que no tenían acceso a los resultados globales y no entendían por qué. Fue necesario redirigirles (vía WhatsApp y móvil).

3.6.8. Modificaciones tras las pruebas

Después de estudiar los resultados de las pruebas se ha decidido realizar las siguientes modificaciones:

- En "Poner notas" controlar que solo se permita la entrada de números con dos decimales.
- En "Poner comentarios"
 - Bloquear los campos después de guardar comentarios. Así quedará claro que se ha finalizado el proceso.
 - Al cambiar de alumno que se reinicie el formulario.
- En "Seguimiento de un alumno", para resaltar la punta de flecha que despliega el detalle, se le pondrá al lado el texto "Ver detalle"
- En el menú inicial, el error de entrar en "Ver notas" (opción de profesor), en lugar de hacerlo en "Seguimiento del alumno", se debe a que ante la pregunta ¿Qué notas tiene ...? el usuario tiende a hacer clic en la primera opción en que aparece, o sea, "Ver notas", sin leer más. Esto es más fácil que ocurra en un móvil, porque para llegar a la opción correcta hay que hacer un gran desplazamiento vertical al final de la pantalla. Para solucionarlo sin cambiar de nuevo el menú, simplemente se cambiará el orden de las opciones, de forma que se vea primero la parte del tutor (y facilite su elección) y después la del profesor.

- Finalmente, se ha observado que la opción de Ayuda o Petición de soporte no se había incluido. Se incluirá como un botón en la barra superior de la aplicación que lance una pantalla informando los datos del administrador.

La versión definitiva de esta aplicación se incluye como una de las partes entregadas en este trabajo, pero para comparar las versiones del producto y observar los cambios hechos tras esta iteración, se puede acceder a las URL:

<http://www.rafaelsoriano-uoc-tfg.com/v1> (antes de esta iteración)

<http://www.rafaelsoriano-uoc-tfg.com/> (después de esta iteración)

Este dominio se mantendrá disponible como mínimo durante el presente mes, enero del 2020.

4. Implementación

4.1. Lenguajes de programación y API utilizados

A fin de llevar a cabo la implementación, desde prácticamente el inicio del proyecto, en paralelo con el desarrollo de las distintas fases de este se realizó una labor de estudio y (re)aprendizaje de las tecnologías disponibles. La primera opción elegida fue Google AppMaker, básicamente por su sencillez y porque muchos centros educativos disponen de Google for Education con una base de datos de alumnos y profesores ya creada y mantenida y que se pensaba reutilizar. Sin embargo, esta herramienta se apoya para la gestión de las bases de datos en Google Cloud Platform, con un coste asociado al consumo de datos y que tras transcurrir el primer mes de uso resultó ser mayor de lo que se preveía que iba a ser y se pensaba soportar.

Descartada la opción Google, se optó por un pack clásico de lenguajes para desarrollo web:

- HTML 5 para la estructura de la página
- CSS 3 para el aspecto
- JavaScript para el comportamiento en el lado del cliente
- PHP para interaccionar con la base de datos en el servidor
- SQL embebido en el PHP para obtener resultados de la base de datos

Además de estos lenguajes se han utilizado las siguientes librerías

- Bootstrap 4. Creado por el equipo de Twitter [2], es un framework para la creación de páginas responsive. Facilita muchísimo el trabajo y enriquece el resultado.
- Normalizer. Librería CSS para garantizar la apariencia homogénea en todos los navegadores
- Modernizr. Librería que permite utilizar navegadores antiguos
- FontAwesome. Es una tipografía que en lugar de estar formada por letras está formada por iconos. Muy útil para botones, menús, ...

4.2. Proceso de programación

De inicio, para establecer la arquitectura de programación de nuestra aplicación vamos a diferenciar tres escenarios

- La interfaz gráfica que verá el usuario. Es lo que llamaremos la **vista**
- El acceso a la base de datos. Este acceso debería estar muy controlado, por eso conviene centralizarlo en un punto. Es lo que llamaremos el **modelo**.
- Las operaciones. Tomarán los datos que provee la base de datos y los manipularán para ofrecerle al usuario los resultados. Es la parte que corresponde a lo que llamamos **controlador**.

Esta forma de estructurar una aplicación se corresponde con un patrón de uso habitual en el desarrollo web, el patrón Modelo-Vista-Controlador [16], que permite desacoplar la interfaz gráfica del resto del sistema, haciendo que el mantenimiento de esta interfaz sea menos costoso.

Este punto es importante ya que una vez establecidas gracias al DCU las funciones que tiene la aplicación, las mejoras en UX vendrán principalmente de modificaciones en la apariencia, convirtiendo a la interfaz de usuario en la parte más propensa a cambios en la fase de programación.

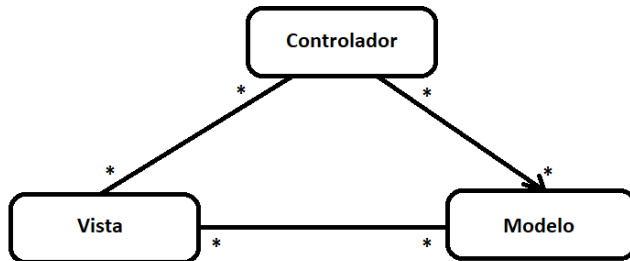


Ilustración 45 Representación gráfica del patrón Modelo-Vista-Controlador

Para mostrar el uso del patrón y la arquitectura de programación resultante, se adjunta un fichero de cada tipo donde se ve claramente su diferencia y también una captura de pantalla con la estructura de carpetas de la web. Si se necesita acceder a la programación completa, se adjunta junto con esta memoria y el resto de los documentos un archivo zip con la aplicación completa.

4.2.1. Fichero Profesores. Ejemplo de archivo perteneciente al modelo

Los archivos que pertenecen al modelo simplemente declaran una clase por cada tabla, que será la que accederá a la BD y los métodos que necesitemos según el tipo de acceso. Por ejemplo, en el caso de la tabla de profesores:

```

/core/models/profesor/profesor.class.php
<?php

//declaramos la clase Profesor para acceder a la tabla Profesores
class Profesor {
//método para recuperar el profesor con el mail y password entrados en el login
public function getByEmailAndPassword($connection, $email, $password){
    try{
        $result = $connection->query("SELECT * FROM profesores WHERE
email = " . $email . " AND password = " . $password . "");
        return $result;
    }catch(Exception $error){
        print $error->getMessage() . "\n";
    }
}

//método para saber si un profesor es tutor y puede acceder a más funciones
public function isTutor($connection, $id){
    try{
        $result = $connection->query("SELECT COUNT(*) AS contadorGrupos
FROM grupos WHERE id_profesor = " . $id);
        //Si el resultado no es vacío, es tutor
        if($result[0]["contadorGrupos"] != 0){
            return 1;
        } else{
            return 0;
        }
    }catch(Exception $error){
        print $error->getMessage() . "\n";
    }
}
}

```

```

    }
  }
}
?>

```

4.2.2. Fichero login. Ejemplo de archivo perteneciente a los controladores

Los archivos tipo controlador manipulan los datos y obtienen resultados, pero siempre acceden a los datos solo a través de las clases del modelo

```

/core/controllers/login/login.php
<?php
    //Iniciamos el uso de sesiones
    session_start();
    //Incluimos el fichero properties
    require_once("../properties.php");
    //Declaramos una variable tipo matriz donde se guardará el usuario recuperado de la BD
    $profesores = Array();
    //Declaramos una variable para guardar el email entrado en la pantalla de identificación (login)
    $email = $_POST["email"];
    //Declaramos una variable para guardar el password entrado en la pantalla de identificación (login)
    $password = $_POST["password"];
    //Instanciamos una conexión a la BD y un objeto de la clase Profesor
    $conexionObject = new DBConnection();
    $profesorObject = new Profesor();
    // Realizamos la consulta
    $profesores = $profesorObject->getEmailAndPassword($conexionObject, $email, $password);
    // Si hay un profesor con esas credenciales, guardamos sus valores en la sesión
    if(count($profesores) == 1){
        $_SESSION["nombreUsuario"] = $profesores[0]["nombre"];
        $_SESSION["idUsuario"] = $profesores[0]["id"];
        $esTutor = $profesorObject->IsTutor($conexionObject, $_SESSION["idUsuario"]);
        $_SESSION["esTutor"] = $esTutor;
        //Y cerramos la conexión a la BD
        $conexionObject->close();
        //Volvemos a la página index.php con la sesión establecida
        header("Location: ../../index.php");
    }
    //Si no hay profesor, los datos de usuario/contraseña entrados no son correctos
    else{
        if(count($profesores) == 0){
            // Cerramos la conexión
            $conexionObject->close();
            //Volvemos a cargar la página de identificación dando error de usuario/contraseña
            header("Location: ../../login.php?mensaje=1");
        }
        // Si hay más de un profesor con esos datos, hay un error en la BD
        else {
            //Cerramos la conexión
            $conexionObject->close();
            //Volvemos a cargar la página de identificación dando error de BD
            header("Location: ../../login.php?mensaje=2");
        }
    }
}
?>

```

4.2.3. Fichero Identificación. Ejemplo de archivo perteneciente a las vistas

Vemos que se trata básicamente de un archivo html que emplea datos que recoge de la sesión que ha rellenado previamente el controlador

```

/identificacion.php (en realidad /login.php)
<?php
    //Include de la cabecera
    require_once('includes/header.php');
?>
    <div class="container login">
        //fuente FontAwesome con el muñequito de alumno con birrete
        <span><i class="fas fa-user-graduate"></i></span>
        <div class="row">
            <div class="col-12">
                <h2>Identificación</h2>
            </div>
        </div>
    </div>

<?php
    // Si tenemos un mensaje de error de usuario, lo avisará aquí
    if (isset($_GET["mensaje"]) && $_GET["mensaje"] == 1){
        echo "<div class='alert alert-danger' role='alert'>";
        echo "El usuario no es correcto";
        echo "</div>";
    }
    else if (isset($_GET["mensaje"]) && $_GET["mensaje"] == 2){
        echo "<div class='alert alert-danger' role='alert'>";
        echo "Error de usuario. Por favor, póngase en contacto con su administrador";
        echo "</div>";
    }
?>
    <form action="core/controllers/login/login.php" method="POST">
        <div class="form-group">
            <input class="form-control" type="text" name="email" placeholder="email" />
            <br />
            <input class="form-control" type="password" name="password"
                placeholder="password" />
            <br />
            <input class="btn btn-primary btn-lg btn-block" type="submit" value="Entrar" />
        </div>
    </form>
</div>
</div>
</div>
<?php
    //Include del pie
    require_once('includes/footer.php');
?>
</body>
</html>

```


4.2.4. Estructura de carpetas de la aplicación

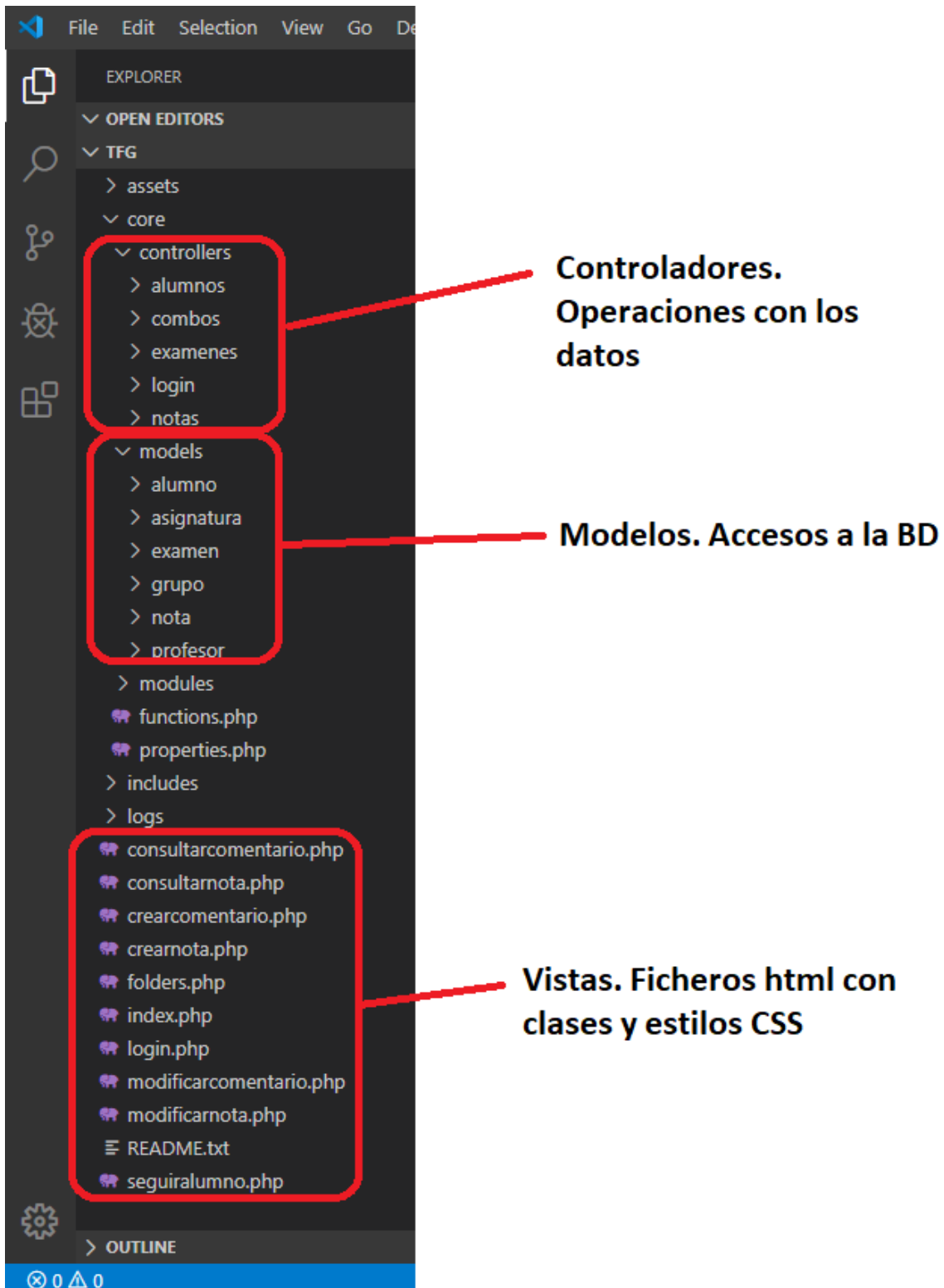


Ilustración 46 Estructura de carpetas de la aplicación reflejando la estructura del patrón MVC

4.3. Requisitos de instalación

Para instalar la aplicación en un ordenador personal se requiere disponer de:

- Servidor de páginas web Apache
- Gestor de base de datos MySQL/MariaDB
- PHP como lenguaje de programación

Todas estas herramientas son de código abierto y su instalación conjunta permite convertir nuestro ordenador en un servidor de páginas o aplicaciones web dinámicas. Este pack de herramientas es muy fácil de instalar, descargando directamente de Internet un paquete LAMP (Linux/Apache/MySQL/PHP) si nuestro sistema operativo es Linux, o bien un paquete WAMP (Windows/Apache/MySQL/PHP) si nuestro sistema operativo es Windows.

Si se requiere acceso público, la mejor opción es contratar un hosting que ya nos ofrezca estas herramientas.

4.4. Instrucciones de instalación

La aplicación viene comprimida en un fichero .zip que al descomprimirlo nos creará una carpeta llamada “ayudaAITutor”. Dentro de esta carpeta tenemos una nueva carpeta y un fichero con extensión .sql que contienen respectivamente la aplicación y la base de datos.

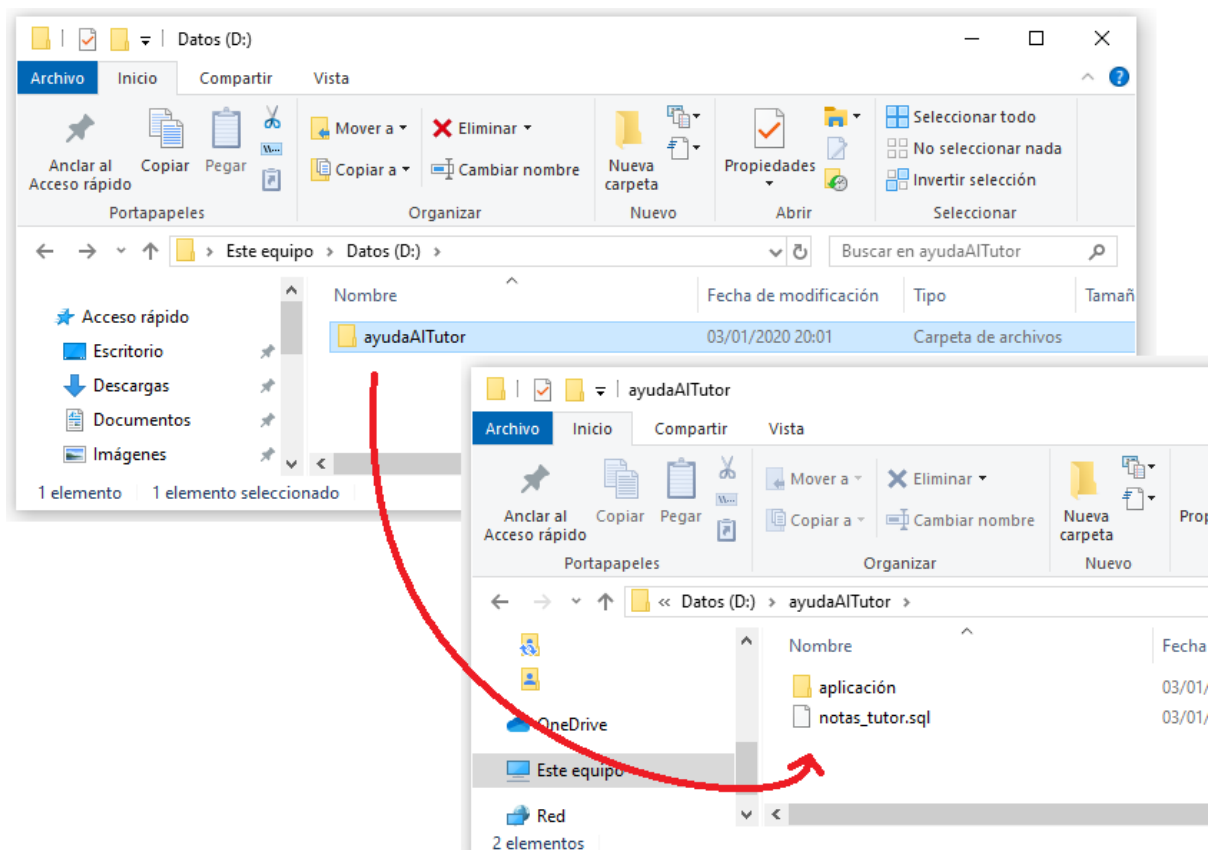


Ilustración 47 Ficheros de la aplicación

Para instalar la aplicación en un hosting contratado basta con copiar todo el contenido de la carpeta “**aplicacion**” (¡ojo, el contenido, no la carpeta!) y pegarlo en la carpeta “de publicación” del servidor web. El nombre de esta carpeta pública puede variar, dependiendo de la configuración, pero algunos nombres candidatos típicos son: htdocs, public, www, o html. El proceso para copiar dependerá del hosting elegido, aunque lo más habitual es hacerlo por ftp.

Si instalamos en nuestro propio equipo, con un LAMP o similar ya instalado, el proceso es el mismo, pero es mejor en este caso copiar la carpeta y no directamente su contenido, ya que la carpeta raíz de publicación suele contener una página de información y enlaces a herramientas necesarias. Para poder ver la aplicación en este caso, en el navegador escribiremos <http://localhost/aplicacion>

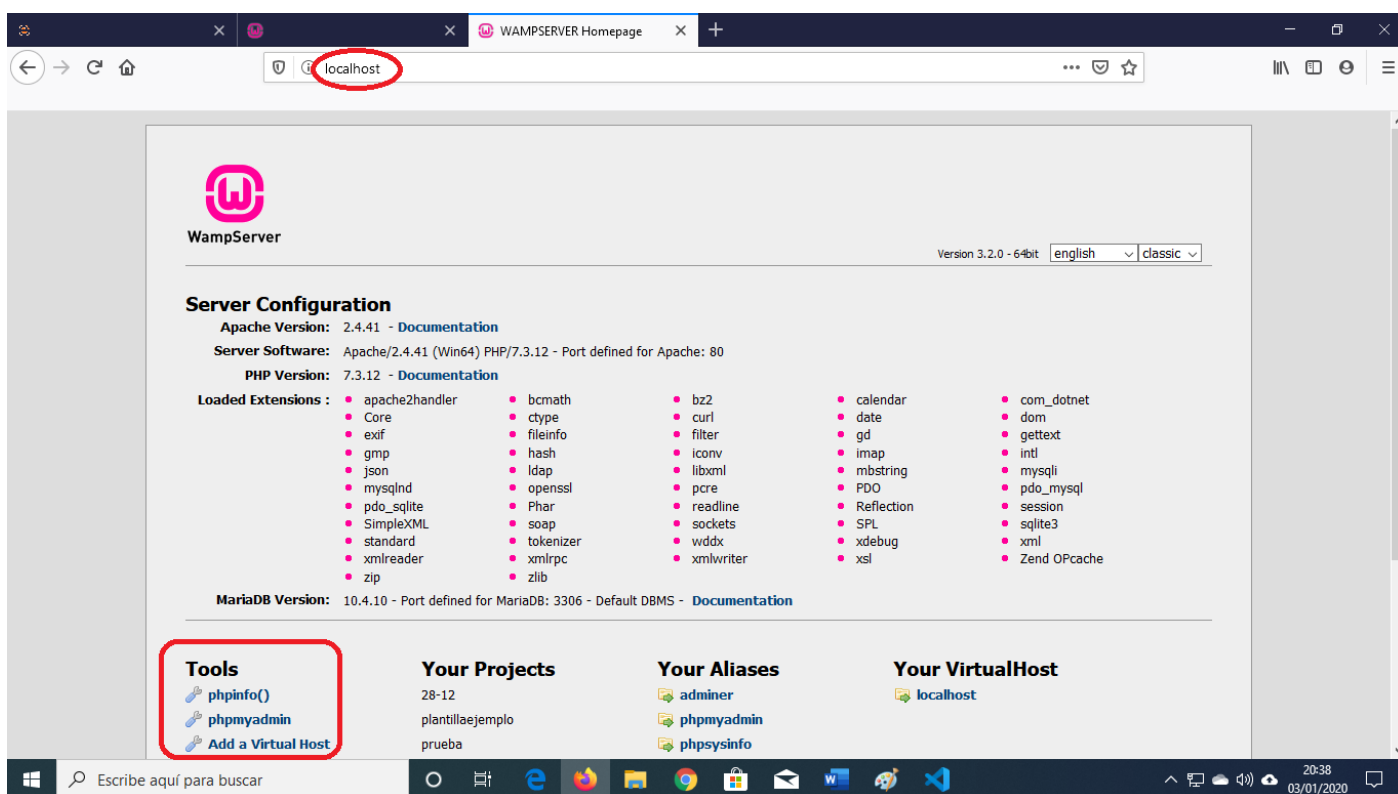


Ilustración 48 Página raíz (localhost) en instalación local de un WAMP

Una vez copiada la carpeta importaremos la BD. Para ello, hay que abrir **phpmyadmin**, en el hosting según nos indique el proveedor, y en local haciendo clic en su enlace, que en este caso tenemos en la página de inicio, abajo a la izquierda (ver ilustración anterior).

Una vez en la aplicación iremos a la pestaña de **Importar**, elegiremos el fichero “**notas_tutor.sql**” y finalmente clic en el botón **Continuar**. Aceptamos los valores por defecto y si todo ha ido bien, al terminar habremos importado la base de datos.

Podemos ver las indicaciones en la figura siguiente:

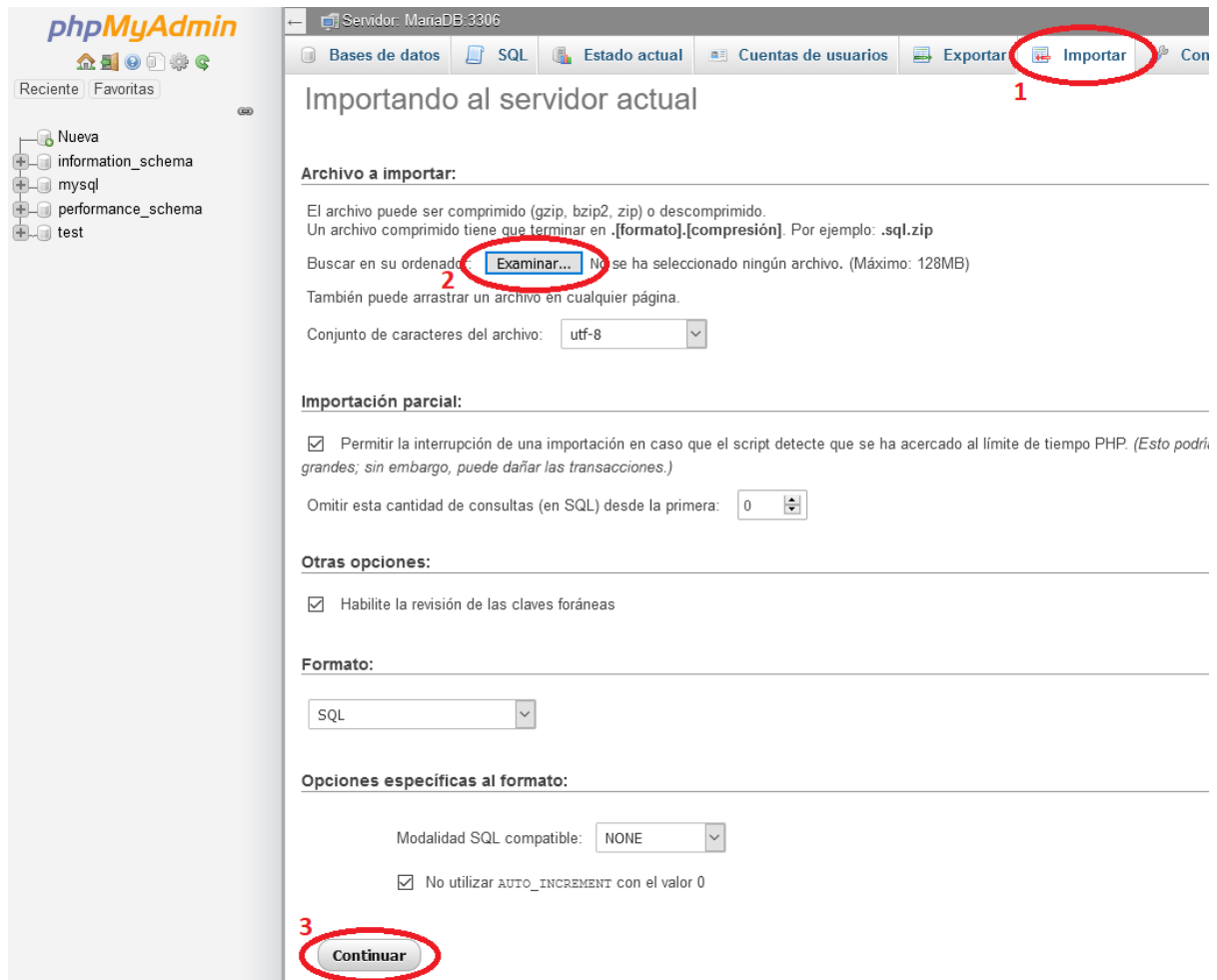


Ilustración 49 Importando la base de datos

Queda un último paso: para acceder a la base de datos necesitamos usuario y contraseña. Nos sirven los mismos que hemos utilizado para entrar en **phpmyadmin**. Para que la aplicación funcione debemos informar esos datos, la forma de hacerlo es escribirlos en el fichero "**properties.php**", que se encuentra dentro de la carpeta "**aplicacion**" en la subcarpeta "**core**".

A continuación, se adjunta el contenido del fichero "**properties.php**" y se indican los valores a sustituir:

```
<?php
// Datos de acceso a BBDD en localhost
//$GLOBALS["host"] = "localhost";
//$GLOBALS["database"] = "notas_tutor";
//$GLOBALS["DBuser"] = "root";
//$GLOBALS["DBpassword"] = "A123456a";

// Datos de acceso en entorno REAL
$GLOBALS["host"] = "qact095.rafaelsoriano-uoc-tfg.com";
$GLOBALS["database"] = "qact095";
$GLOBALS["DBuser"] = "qact095";
$GLOBALS["DBpassword"] = "A123456a";

// Cargamos la librería de funciones
require_once("../functions.php");
```

```
// Cargamos la librería de Clases
require_once ("../../modules/DBConnection/DBConnection.class.php");

require_once ("../../models/profesor/profesor.class.php");
require_once ("../../models/grupo/grupo.class.php");
require_once ("../../models/asignatura/asignatura.class.php");
require_once ("../../models/alumno/alumno.class.php");
require_once ("../../models/examen/examen.class.php");
require_once ("../../models/nota/nota.class.php");

?>
```

5. Conclusiones

En general los informáticos somos un gremio con una asignatura pendiente con la sociedad. Nos cuesta que nos entiendan y para constatarlo basta con observar el cliché con que nos catalogan normalmente: frikis. Siempre he estado en contra de este estereotipo, y al hablar con un usuario jamás empleo jerga si no es estrictamente necesario. Partiendo de esta premisa, en mi elección de área para el TFG no tuve dudas: Interacción Persona Ordenador. Es un área que me parece fundamental. Ahora, finalizando mi proyecto, creo que acerté y se ha reforzado mi convencimiento. Dentro de esta área, para desarrollar el trabajo se utiliza la metodología DCU. Yo la desconocía, y al estudiarla y aplicarla me ha parecido de una claridad, una eficacia y una sensatez descomunales. Además, seguir sus fases ha sido fundamental para poder avanzar en la planificación y consecución final de los objetivos. Dentro de la metodología DCU ha sido especialmente enriquecedor todo el trabajo de entrevistas y pruebas con usuarios

En cuanto al alcance del proyecto, mi decisión de llegar hasta un producto real totalmente funcional e implementado me plantea dos reflexiones. De un lado, la implementación de una aplicación, por simple que sea, conlleva un esfuerzo que me ha hecho dudar de si conseguiría mi objetivo final. Creo honestamente que no fue buena idea en una asignatura tan importante y que ya implica una carga extra de trabajo, optar por esta opción, y no se la recomendaría a un estudiante a punto de empezar su TFG. Sin embargo, por otro lado, llegar hasta la implementación me ha aportado puntos de vista más completos y enriquecedores, y me ha permitido conocer la programación web responsive que hoy en día ocupa una posición importante en el mercado, además de practicar un abanico variopinto y laborioso de tareas, como, por ejemplo, generar a través de webs de objetivos dudosos, nombres y apellidos aleatorios para alimentar la base de datos.

Con respecto al producto conseguido, considero que si bien, tiene campo de mejora, los objetivos están cumplidos, la aplicación funciona y sin duda, como su nombre indica, ayuda al tutor en su trabajo. Teníamos una necesidad y hemos diseñado una solución. De hecho, en las últimas pruebas de usuario realizadas, ya con la aplicación real, el comentario de “Oye, que no quede solo en TFG, estaría muy bien que lo usásemos” se me ha regalado en varias ocasiones.

Bibliografía

- [1] Additio App. Cuaderno de notas para el profesor <https://www.additioapp.com/es/> (última visita 14/10/2019)
- [2] Bootstrap The most popular HTML, CSS, and JS library in the world [en línea] <https://getbootstrap.com/> (última visita 12/12/2019)
- [3] **Carraro, Juan Manuel**. Iterando: Como realizar un test de usabilidad en cinco pasos [en línea] <http://www.iterando.com.ar/2008/03/como-realizar-un-test-de-usabilidad-en.html> (última visita 1/12/2019)
- [4] **Casado Martínez, Carlos; Garreta Domingo, Muriel; Hassan Montero, Yusef; Martínez Normand, Loïc; Mor Pera, Enric**. Interacción Persona Ordenador. UOC. PID 00180441. ISBN 9788469342237 [en línea] http://materials.cv.uoc.edu/daisy/Materials/PID_00158925/html5/index.html (última visita 23/11/2019)
- [5] **Castilla Romero, Claudio** (2012) GUÍA DEL TUTOR DE SECUNDARIA – ISBN 978-84-695-3708-4 – Consejería de Educación, Formación y Empleo. MURCIA
- [6] Design Toolkit. Arquitectura de la información [en línea] <http://design-toolkit.recursos.uoc.edu/es/guia/arquitectura-de-la-informacion/> (última visita 18/11/2019)
- [7] Design Toolkit. Card Sorting [en línea] <http://design-toolkit.recursos.uoc.edu/es/guia/card-sorting/> (última visita 19/11/2019)
- [8] Design Toolkit. Diagramas de flujo [en línea] <http://design-toolkit.recursos.uoc.edu/es/diagrama-de-flujo/> (última visita 28/11/2019)
- [9] Design Toolkit. Prototipado [en línea] <http://design-toolkit.recursos.uoc.edu/es/prototipado/> (última visita 01/12/2019)
- [10] GESTIB. Govern de les Illes Balears <https://www3.caib.es/xestib/?idioma=es> (última visita 20/12/2019)

[11] **Gil Ripoll, David.** Metodología de trabajo DCU [en línea] <https://es.slideshare.net/KingeClient/metodologa-dcu-diseo-centrado-en-el-usuario> (última visita 29/12/2019)

[12] Idoceo <https://www.idoceo.es/index.php/es/> (última visita 14/10/2019)

[13] INFORME DEL PROFESORADO AL TUTOR/A EDITABLE. Para preparar entrevista con padres. <https://www.orientacionandujar.es/2017/08/28/informe-del-profesorado-al-tutora-editable-preparar-entrevista-padres/> (última visita 14/10/2019)

[14] JQuery [en línea] <https://jquery.com/> (última visita 12/12/2019)

[15] Plataforma de gestión para colegios. Clickedu <https://clickartedu.com/inicio-plataforma-colegios.html> (última visita 14/10/2019)

[16] **Pradel i Miquel, Jordi; Raya Martos, José Antonio.** Catálogo de patrones. Módulo 2 de la asignatura Análisis y diseño con patrones -UOC- [en línea] http://cv.uoc.edu/annotation/633cfcef4bf7d13addbab2f0450d4111/526372/PID_00160148/modul_2.htm#w26aab9c15c11 (última visita 29/12/2019)

[17] SOFTAULA plataforma CRM/ERP de gestión escolar <https://softaula.net/> (última visita 14/10/2019)

[18] **Travis, David** User-Centered Design: The Fable of the User-Centered Designer [en línea] <https://www.userfocus.co.uk/fable/index.html> (última visita 19/11/2019)

[19] Wikipedia. Diseño web adaptable [en línea] https://es.wikipedia.org/wiki/Dise%C3%B1o_web_adaptable (última visita 28/12/2019)

Anexo 1: Hoja resumen de las entrevistas

Situación actual.

Pregunta: ¿Qué sistema sigues para hacer seguimiento y recopilar las notas de todas las asignaturas?

Curso	Respuesta
1º ESO A	
1º ESO B	Perseguir a los profesores implicados. Consultar a los niños.
2º ESO A	Rellenar la hoja de seguimiento y dejarla sobre la mesa central de la sala de profesores. Comentárselo a los profesores implicados para que la vean y la rellenen. Recogerla el día antes de la entrevista
2º ESO B	
3º ESO A	
3º ESO B	Básicamente hacer seguimiento pidiéndoselo a los niños. Si en algún caso hay sospechas de que no son las correctas, sondear a algunos profes.
4º ESO A	Perseguir a los profesores implicados para rellenar el formulario y pedírselas a los chicos
4º ESO B	Creación de un grupo de difusión de WhatsApp
1º Bach A	Hoja de seguimiento y WhatsApp
1º Bach B	Ir consultando con los profesores con los que se coincide ocasionalmente
2º Bach A	
2º Bach B	

Conclusiones:

- Es llamativo que bastantes profesores utilizan el término perseguir. Aunque el ambiente de la entrevista es coloquial y relajado, da idea de la dificultad que implica contactar entre profesores. Otra muestra de esto es esta misma entrevista, donde después de cerca de dos semanas aún quedan muchos huecos por rellenar.
- El procedimiento principal se basa en rellenar la hoja establecida.
- En muchos casos, sobre todo si hay cercanía a nivel personal, se utiliza el WhatsApp, pese a estar desaconsejado por temas de protección de datos.
- Algunos tutores, aprovechan la hora de tutoría para solicitar a todos los alumnos las notas obtenidas a lo largo de la semana. De esta manera mantienen un registro bastante actualizado, aunque sujeto a ciertas reservas en cuanto a su veracidad (en ocasiones, el adolescente puede “mejorar” una nota para evitar una llamada a casa).

Características deseables en la nueva aplicación

Pregunta: Si se desarrolla una nueva aplicación para facilitar esta tarea, ¿qué información sería más relevante? ¿Cuál sería un formato adecuado para mostrar los datos? ¿Qué características serían deseables? ¿Otras propuestas?

Curso	Respuesta
1º ESO A	
1º ESO B	Que sea multiplataforma, en particular, accesible vía ordenador (el profesor entrevistado no utiliza en su día a día móvil). Que sea posible ver solo los que no tienen media de aprobado
2º ESO A	No sabe que decir
2º ESO B	
3º ESO A	
3º ESO B	Que no sea complicado
4º ESO A	Accesible en Tablet y en móvil. Que los profesores no tutores solo puedan ver las notas de su clase. Estaría bien forzar a que sea herramienta oficial del centro, así se usaría seguro.
4º ESO B	Que se puedan añadir todo tipo de comentarios. Adjuntar una foto para no tener que entrar las notas. Que funcione en el móvil
1º Bach A	No sabe que decir
1º Bach B	No sabe que decir
2º Bach A	
2º Bach B	

Conclusiones:

- Desde el punto de vista del entrevistador, no parece adecuado influir o incitar la respuesta, como por ejemplo sucede si se hacen preguntas del tipo “¿Qué te parecería si la aplicación incluyera tal funcionalidad?” Se han evitado este tipo de preguntas.
- Al actuar de este modo y encontrarnos en un punto tan inicial del proyecto, las aportaciones de los usuarios en esta segunda parte de la entrevista han sido poco concluyentes o, directamente, no han sabido que aportar en la mayoría de los casos.
- Se sugiere la idea de incluir comentarios y no solo notas. También se pone de manifiesto la necesidad de hacer un producto multiplataforma. Por lo demás, no se considera que haya habido aportaciones interesantes y novedosas que no se tuvieran ya presentes en el proyecto. Probablemente, en las pruebas de usabilidad, con un prototipo de producto concreto en manos de los usuarios, estos sí serán capaces de hacer aportaciones valiosas para mejorar la aplicación.

Anexo 2: Modelo oficial de recogida de notas en el centro de referencia del TFG



FULL DE TUTORIA. ESO. 1R CICLE. CURS ACADÈMIC 20__ 20__.

ALUMNE/A: _____ CURS _____ GRUP _____

DIA DE VISITA: _____ MOTIU DE VISITA: _____

RESULTATS ACADÈMICS I OBSERVACIONS DEL PROFESSORAT.

C. NATURALS.....

ACTITUDS _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

C. SOCIALS.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

E. FÍSICA.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

ED. PLÀS. I VISUAL.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

LL. ESPANYOLA.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____



LL. CATALANA.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

ANGLÈS.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

MATEMÀTIQUES.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

MÚSICA.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

TECNOLOGIA.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

FRANCÈS.....

ACTITUD _____

CONEIXEMENTS _____

POSSIBLES SOLUCIONS _____

Ilustración 51 Modelo oficial de recogida de notas. Hoja 2



INFORMÀTICA.....

ACTITUD

CONEIXEMENTS

POSSIBLES SOLUCIONS

PROC. DE COMUNICAC.

ACTITUD

CONEIXEMENTS

POSSIBLES SOLUCIONS

RELIGIÓ.....

ACTITUD

CONEIXEMENTS

POSSIBLES SOLUCIONS

OBSERVACIONS.

Ilustración 52 Modelo oficial de recogida de notas. Hoja 3

Anexo 3: Lista de profesores cargados en la aplicación para poder hacer pruebas

Usuario	Contraseña	Tutor
aguasch@micolegio.es	123456	1º ESO B
sribes@micolegio.es	123456	1º ESO A
lcepeda@micolegio.es	123456	1º Bach A
lmarin@micolegio.es	123456	4º ESO A
jramirez@micolegio.es	123456	2º Bach B
aesquivel@micolegio.es	123456	3º ESO B
rpelayo@micolegio.es	123456	
padame@micolegio.es	123456	
fmayor@micolegio.es	123456	
msantacruz@micolegio.es	123456	
mleal@micolegio.es	123456	
ccantó@micolegio.es	123456	2º Bach A
lquesada@micolegio.es	123456	
raranda@micolegio.es	123456	2º ESO B
nmelgar@micolegio.es	123456	2º ESO A
jleon@micolegio.es	123456	4º ESO B
acarrera@micolegio.es	123456	
kgastelu@micolegio.es	123456	
jcastaneda@micolegio.es	123456	1º Bach B
aarreola@micolegio.es	123456	3º ESO A
ntapia@micolegio.es	123456	
igaray@micolegio.es	123456	
ibetancourt@micolegio.es	123456	
fpineda@micolegio.es	123456	
nestevez@micolegio.es	123456	
mros@micolegio.es	123456	
mripoll@micolegio.es	123456	

Tabla 7 Listado de usuarios (profesores) válidos para probar la aplicación