

Grado en Ingeniería de Sistemas Audiovisuales
2018-2019

Trabajo Fin de Grado

“Sistema de soporte a la interacción
basado en datos para promover el
aprendizaje activo y colaborativo”

Sergio Fernández Sánchez

Tutora

Raquel M. Crespo García

Leganés, 2019



Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento**
– **No Comercial** – **Sin Obra Derivada**

DEDICATORIA

Primero de todo me gustaría agradecer a mi tutora por ofrecerme este Trabajo de Fin de Grado y por haberme prestado su ayuda durante todo el proceso.

Quiero agradecer a mis padres por todo lo que me han dado en la vida, que es innumerable. Gracias por estar ahí siempre y por haberme apoyado tanto durante mi etapa universitaria. Ha merecido la pena el camino, gracias por llevarme hasta aquí.

Gracias a mi pareja, Patri, por el apoyo incondicional todos estos años. Gracias por todos los ánimos en cada momento duro, gracias por ser tan comprensiva siempre conmigo y por valorarme tanto. Gracias por estar todo este tiempo a mi lado y hacerme tan feliz.

Agradecer también a toda mi familia y amigos, por todo el apoyo durante esta etapa que ya está llegando a su fin. Gracias a mis compañeros Pedro, Otero e Irina, por haber hecho el camino mucho más llevadero y por haberme ayudado en tantas ocasiones.

Y por último, gracias a mis primos, a mi tía Nieves y en especial a mi tío Ismael, por ser un ejemplo de lucha y valentía durante este tiempo y haberme apoyado todos estos años. Te echamos mucho de menos Ismael.

RESUMEN

En este proyecto se hará uso de la tecnología para tratar de mejorar la experiencia en la enseñanza del profesor y los alumnos. Se busca dar solución a dos problemas que surgen a menudo en las sesiones prácticas; la gestión de las dudas y la falta de información en tiempo real para el profesor. Para ello, se diseña ClassOn, una aplicación web que incluye las siguientes funcionalidades:

- Creación de un sistema de gestión de dudas que permita a los alumnos subir sus dudas a la aplicación y facilite al profesor la selección del orden en el que atiende a los alumnos.
- Creación de un mapa de la clase con información de los alumnos que se encuentran en ella, proporcionando así al profesor la visualización de su progreso en tiempo real.

El objetivo de este proyecto es terminar el desarrollo de la última versión de ClassOn, consiguiendo que la aplicación incluya las funcionalidades descritas y que sea una herramienta intuitiva, efectiva y adaptativa para distintos dispositivos.

INDICE DE CONTENIDO

1. INTRODUCCIÓN	1
1.1. Motivación del proyecto	1
1.2. Objetivo del proyecto.....	2
1.3. Estructura de este documento	3
2. ESTADO DEL ARTE	5
2.1. Metodologías de enseñanza	5
2.2. Soluciones tecnológicas en la enseñanza.....	6
2.3. Backchannel.....	7
2.3.1. Objetivo.....	7
2.3.2. Aplicaciones relacionadas	8
2.3.3. Problemas	10
2.4. Visualización de datos	10
2.4.1. Objetivo.....	10
2.4.2. Aplicaciones relacionadas	11
2.4.3. Problemas	13
2.5. Versiones anteriores de ClassOn	13
2.6. Estado del arte de la tecnología utilizada.....	17
3. DESCRIPCIÓN FUNCIONAL	19
3.1. Introducción	19
3.2. Objetivo de la aplicación	19
3.3. Estructura básica	20
3.3.1. Bloque del profesor	20
3.3.2. Bloque del alumno	22
3.4. Backchannel en ClassOn	23
3.5. Visualización de datos en ClassOn.....	25

3.6.	Consideraciones técnicas previas.....	26
4.	IMPLEMENTACIÓN DE LA SOLUCIÓN	27
4.1	Introducción	27
4.2.	Arquitectura tecnológica.....	27
4.2.1	Back-end.....	28
4.2.2.	Front-end	29
4.2.3.	Estructura del código.....	30
4.3.	Descripción de las pantallas y sus funcionalidades	31
4.3.1.	Componentes genéricos.....	31
4.3.2.	Pantalla inicial	32
4.3.3.	Registro	33
4.3.4.	Inicio de sesión.....	35
4.3.5	Panel del profesor.....	36
4.3.6	Creación / selección de una clase	36
4.3.7.	Mapa de la clase	38
4.3.8.	Creación de tareas	43
4.3.9.	Panel del estudiante	44
4.3.10.	Vista de tareas del estudiante	46
5.	GESTIÓN DEL PROYECTO	49
5.1.	Planificación del proyecto	49
5.2.	Tecnología y recursos requeridos	50
5.3.	Presupuesto	51
5.4.	Modelo de negocio.....	52
5.5.	Entorno socio-económico	53
6.	MARCO REGULADOR.....	54
7.	CONCLUSIONES Y TRABAJO FUTURO	56
7.1.	Conclusiones del proyecto	56

7.2. Trabajo futuro	56
BIBLIOGRAFÍA	58
ANEXO A – TÉRMINOS LEGALES	
ANEXO B – ENGLISH SUMMARY	

ÍNDICE DE FIGURAS

Figura 2.1. Diseño de la interfaz para alumnos de la versión de 2011	14
Figura 2.2. Diseño de la interfaz para el profesor de la versión de 2011	15
Figura 2.3. Arquitectura de las tecnologías utilizadas en la versión de 2012.....	15
Figura 2.4. Interfaz del profesor en tablets para un escenario con muchos alumnos	16
Figura 3.1. Diagrama de flujo de la gestión de dudas	24
Figura 4.1. Esquema de la arquitectura tecnológica de ClassOn.....	27
Figura 4.2. Barra de navegación.....	31
Figura 4.3. Diseño de la pestaña en el navegador Chrome.....	32
Figura 4.4. Pantalla inicial de la aplicación.....	32
Figura 4.5. Pantalla de registro para estudiantes y profesores.....	34
Figura 4.6. Pantalla de inicio de sesión para estudiantes y profesores	35
Figura 4.7. Panel del profesor.....	36
Figura 4.8. Pantalla de creación de una clase nueva	37
Figura 4.9. Mapa de una clase de 4x5 con ocho alumnos en ella.....	38
Figura 4.10. Tarjetas en función de su ancho	39
Figura 4.11. Tarjetas en función de su altura	40
Figura 4.12. Ejemplo de mapa con un caso límite.....	41
Figura 4.13. Información detallada de un puesto con dos alumnos	41
Figura 4.14. Tablón de dudas debajo del mapa de la clase	42
Figura 4.15. Pantalla para añadir secciones a una tarea	44
Figura 4.16. Recorte de la pantalla con el panel del estudiante.....	45
Figura 4.17. Pantalla en la que el estudiante selecciona su puesto.....	45
Figura 4.18. Pantalla de la vista de tareas para el estudiante.....	46
Figura 4.19. Tablón de dudas en la vista de tareas del alumno	47
Figura 4.20. Pop-up para que el alumno responda dudas de sus compañeros.....	48

ÍNDICE DE TABLAS

Tabla 5.1. Planificación del proyecto	49
Tabla 5.2. Costes debidos al personal	51
Tabla 5.3. Costes debidos a la depreciación de los equipos	52

1. INTRODUCCIÓN

1.1. Motivación del proyecto

En una época marcada por el progreso tecnológico, la mayoría de los sectores se ven beneficiados con las nuevas herramientas y facilidades que ofrece la tecnología. En el campo de la enseñanza el uso de estos recursos es muy práctico en muchos sentidos, por esto, la presencia de la tecnología en las aulas es cada vez mayor.

En mi etapa de estudiante en la universidad, he tenido distintos tipos de asignaturas, y para muchas de las materias, el único uso de herramientas digitales ha sido el almacenamiento de los temas en Aula Global. ¿Es esto suficiente? ¿Se aprovecha del todo la tecnología que se tiene al alcance?

En muchas asignaturas, las clases se desarrollan en un aula informática donde los alumnos se reparten por los equipos disponibles. En estas clases, en primer lugar el profesor plantea el tema que se va a explicar en esa sesión y puede, o no, realizar una explicación teórica al respecto. Tras esto, los alumnos acceden a distintas tareas que el profesor ha subido al Aula Global y deberán ir realizándolas mientras van entendiendo el contenido de forma autónoma. Llegados a este punto, el profesor se dedica a atender las dudas de los alumnos según estos levantan la mano, y es aquí donde, en la mayoría de los casos, surge el problema.

Cualquier compañero o compañera que lea esto, coincidirá conmigo en que esta fase se puede volver un poco caótica. Los alumnos deben estar pendientes del profesor cada vez que quieren ser atendidos y, por lo general, no se respeta el turno de espera. Muchas veces, esto causa que los alumnos se detengan en el proceso de la tarea o incluso desechen su duda y nunca lleguen a resolverla.

¿Es esto un problema causado por el docente? De ningún modo. El profesor, al atender una duda, pierde inevitablemente la atención de lo que pasa en el resto de la clase, y tiene

que ir de puesto en puesto atendiendo las preguntas lo más rápido posible porque sino se acumulan las manos levantadas del resto de los alumnos.

Este no es el único problema que se puede dar en estas clases. En el lado opuesto, hay muchas veces que los alumnos no tienen dudas pero se atascan en alguna de las tareas. El profesor no es capaz de conocer el progreso de cada uno de los alumnos, por lo que desconoce si muchos alumnos se bloquean en el mismo punto o si hay algunos estudiantes muy rezagados respecto al resto. Esto puede causar que algunos estudiantes no consigan adquirir los conocimientos de dicha sesión y que, a la larga, se pierdan en la asignatura entera.

Ante estos problemas, surge **ClassOn**, una aplicación web que pretende mejorar la experiencia en la enseñanza tanto para el profesor como para los estudiantes. ClassOn busca dar solución a los problemas planteados mediante dos funcionalidades principales:

- Creación de un sistema de gestión de dudas que permita a los alumnos subir sus dudas a la aplicación y facilite al profesor seleccionar el orden en el que atiende a los alumnos. Además, las dudas serán visibles para el resto de los alumnos, pudiendo estos contestar las dudas de sus compañeros.
- Creación de un mapa de la clase con información de los alumnos que se ubican en ella, proporcionando así al profesor la visualización de su progreso en tiempo real.

1.2. Objetivo del proyecto

El objetivo de este proyecto es finalizar el desarrollo de una nueva versión de ClassOn. Para conseguir esto, se pueden distinguir los siguientes sub-objetivos:

- Analizar las metodologías de enseñanza en las que se basa esta aplicación y las soluciones tecnológicas que aplica para llevarlas a cabo.

- Entender los objetivos que persigue esta versión de ClassOn y conocer las versiones anteriores desarrolladas.
- Comprender el funcionamiento del lado del servidor (back-end) que está previamente desarrollado y analizar sus funcionalidades.
- Análisis de las tecnologías utilizadas para el desarrollo de la aplicación.
- Desarrollar la nueva versión de ClassOn que incluya las dos funcionalidades principales descritas previamente: gestión de las dudas y visualización del progreso de los alumnos. Este proyecto se enfoca principalmente al lado del cliente (front-end) pero también será necesario añadir nuevas funcionalidades en el lado del servidor.
- Planificación del proyecto y análisis de su impacto socio-económico.
- Análisis del marco regulador que se aplica a ClassOn.

1.3. Estructura de este documento

Esta memoria se divide en siete capítulos, cada uno dividido a su vez en varios apartados y sub-apartados. A continuación se describirá el contenido de los capítulos por orden de aparición:

Capítulo 1: Introducción. Este capítulo contiene la motivación de la realización de este proyecto, los objetivos que se pretenden conseguir y la estructura del documento.

Capítulo 2: Estado del arte. En este capítulo se describirán las distintas herramientas digitales que se usan en la enseñanza y se hará un análisis más a fondo de las metodologías en las que se basa esta aplicación. También se describirán algunos ejemplos de aplicaciones similares así como las versiones anteriores de ClassOn. Además, se realizará un análisis del estado del arte en las tecnologías utilizadas en el desarrollo.

Capítulo 3: Descripción funcional. En este capítulo se describirá la funcionalidad del sistema que se quiere desarrollar indicando los usos de las soluciones tecnológicas que se

incluyen en la aplicación. Además, se realizará un análisis de las decisiones técnicas que son requeridas para el desarrollo.

Capítulo 4: Implementación de la solución. En este capítulo se describe todo el proceso del desarrollo de la aplicación. Se incluye la arquitectura tecnológica de ClassOn, descripciones del lado del cliente y del servidor y una descripción detallada de cada una de las pantallas de la aplicación con sus funcionalidades.

Capítulo 5: Gestión del proyecto. En este capítulo se incluye la planificación detallada del proyecto, con la descripción de cada fase y su duración. Además, se describen los recursos técnicos requeridos y el entorno socio-económico del proyecto, detallando los gastos, los posibles ingresos futuros y el impacto socio-económico que conlleva la aplicación.

Capítulo 6: Marco regulador. En este capítulo se incorporan los aspectos relativos al marco que regula la aplicación. Se incluye un análisis de la legislación aplicable sobre ClassOn y los estándares técnicos de la tecnologías utilizadas en su desarrollo.

Capítulo 7: Conclusiones y trabajo futuro. En este capítulo se analizará el resultado conseguido respecto al objetivo inicial y se describirá el posible trabajo futuro para las siguientes versiones de la aplicación.

2. ESTADO DEL ARTE

2.1. Metodologías de enseñanza

Una metodología de enseñanza comprende los principios y métodos que usan los docentes para enseñar a los alumnos. Son muchas las posibles metodologías a seguir en la enseñanza y a menudo se emplea la combinación de varias a la vez.

ClassOn orienta su funcionamiento a distintas metodologías de enseñanza que se basan en un aprendizaje participativo, colaborativo y en la interacción profesor-alumno. Estas metodologías son las siguientes:

- **Aprendizaje activo** [1]: metodología que se basa en que el alumno aprenda a través de una participación activa en vez de permanecer pasivo mientras el profesor explica la materia. Este método es beneficioso porque el estudiante se siente involucrado en el aprendizaje, mejorando así la comprensión de los contenidos y desarrollando sus capacidades de pensamiento crítico y resolución de problemas. Como se señala en [2]: “las oportunidades de que los estudiantes aprendan se ven reducidas cuando su papel en las clases es principalmente el de espectadores”.

El aprendizaje activo suele ir combinado con otra metodología, el **aprendizaje colaborativo**, que se basa en dividir a los alumnos en pequeños grupos con el fin de que trabajen juntos para conseguir un objetivo académico común. Con este aprendizaje se busca que todos los miembros del grupo contribuyan al trabajo aportando ideas, colaborando en la resolución de problemas y analizando como llegar a un determinado objetivo.

Hay dos modelos de enseñanza que están basados en estas dos metodologías:

- **Aprendizaje basado en proyectos** (*project-based learning*, **PBL**) [3]: se fundamenta en la realización de proyectos basados en problemas reales. Se busca que los alumnos, divididos en grupos, investiguen y trabajen conjuntamente para conseguir llevar a cabo un proyecto relacionado con

uno o varios temas. El objetivo es que los estudiantes desarrollen competencias de cooperación y comunicación y además adquieran conocimientos sobre la materia.

- **Aprendizaje basado en problemas (*problem-based learning*, PBL)** [4]: similar al aprendizaje basado en proyectos, pero éste se basa en la realización de problemas más cortos y por lo general más guiados. [5]
Los objetivos de este método son mejorar las aptitudes de trabajo en equipo de los alumnos y garantizar que éstos adquieran conocimientos sobre la materia.
- ***Just in time teaching* (JITT)** [6]: metodología que busca adaptar el contenido impartido en las clases utilizando tests o cuestionarios que se realizan al inicio o al final de cada sesión. Con los resultados de estos cuestionarios, el profesor puede orientar las clases de una forma u otra, buscando así el ritmo óptimo para el aprendizaje de los estudiantes. Es una metodología basada en la metodología de aprendizaje activo.
- ***Peer Instruction* (PI)** [7]: metodología que utiliza la iteración directa entre compañeros como medio de aprendizaje. Puede entenderse como una estrategia basada la metodología de aprendizaje colaborativo. Se busca que los alumnos se ayuden unos a otros a entender la materia propuesta y que analicen juntos la explicación del docente mediante el trabajo en pequeños grupos. Con esta metodología se fomenta el trabajo en equipo, se mejora la retención de los conocimientos adquiridos y se comprueba que los estudiantes han entendido bien los conceptos de la lección impartida.

2.2. Soluciones tecnológicas en la enseñanza

Las metodologías descritas en el apartado anterior, se ven beneficiadas con el uso de soluciones tecnológicas. Las herramientas digitales en la enseñanza pueden estar diseñadas con múltiples fines académicos: facilitar la colaboración y la interacción, dar soporte al alumno, optimizar el trabajo del docente, alojar material didáctico, etc.

En este proyecto, se seleccionan soluciones tecnológicas que permiten mejorar la interacción alumno-profesor y promover el aprendizaje colaborativo.

En los siguientes capítulos se va a profundizar acerca de las dos soluciones que se utilizan en ClassOn; *backchannel* y visualización de datos en tiempo real.

2.3. Backchannel

Backchannel es una solución tecnológica que se basa en la creación de un diálogo en tiempo real entre los alumnos y el profesor para que se establezca una conversación paralela a la lección que está siendo impartida. Mediante el uso de un chat o un muro digital, se busca que los alumnos planteen sus dudas, respondan posibles preguntas de sus compañeros o puedan incluso proseguir la conversación sobre la clase después de que ésta termine. [8]

2.3.1. Objetivo

El objetivo del *backchannel* es que mediante este sistema los alumnos interactúen con el profesor sin interrumpir la dinámica habitual de la clase, impulsando así la participación de los alumnos y facilitando la resolución de sus dudas. [9]

Se busca promover un intercambio de información paralelo a la explicación principal que permita a los estudiantes plantear dudas directamente al profesor o publicarlas en un tablón digital donde otros alumnos puedan responderlas.

En clases con muchos estudiantes, también se puede utilizar un sistema de votación para las dudas ya lanzadas, de forma que al docente le lleguen únicamente las dudas más extendidas entre los alumnos.

Mediante el uso de un tablón digital se facilita la colaboración inmediata entre personas que tratan de alcanzar un mismo objetivo. Gracias a esto, se mejora la capacidad de trabajo en equipo de los alumnos, una aptitud que deben desarrollar antes de su llegada al mundo laboral.

2.3.2. Aplicaciones relacionadas

En este apartado se describirán algunas herramientas digitales que utilizan el sistema *backchannel*.

- **Backstage**

Backstage [10] es un software que permite la comunicación entre alumnos y profesor en un segundo plano paralelo a la explicación principal. Su funcionamiento se basa en la creación de un muro digital de micro-mensajes donde toda la clase puede leer y escribir comentarios (opcionalmente de forma anónima). Además, esta herramienta incorpora la posibilidad de que solo se muestren en el muro los mensajes más votados, siendo los alumnos los encargados de decidir que mensajes llegan al profesor.

En una versión posterior [11] añaden una nueva funcionalidad, la posibilidad de almacenar los mensajes del *backchannel* en las propias diapositivas de la materia. Mediante esta funcionalidad, el profesor puede observar las reacciones de los estudiantes en cada sección y los alumnos tienen las respuestas a sus dudas en el lugar donde surgieron. Además, incorporan estadísticas en los perfiles de los estudiantes, de forma que se distingue a los alumnos más y menos activos.

- **Google Classroom**

Google Classroom [12] es una aplicación desarrollada por Google dentro de su programa G Suite para ayudar al profesor y hacer más productivo el aprendizaje de los estudiantes. Se trata una aplicación multi-plataforma desarrollada para Android, iOS y Web que es totalmente gratuita sin ningún tipo de cuenta premium.

Google Classroom tiene la ventaja de contar con muchas herramientas de apoyo para el profesor dentro del programa *G Suite for Education*: Drive, Calendar, Docs, Forms o Groups [13]. Estas herramientas además cuentan con un gran

sistema de soporte y guías para que el profesor optimice las clases y mejore la comunicación con sus alumnos [14].

La aplicación cuenta con un tablón en el que tanto alumnos como profesor pueden añadir publicaciones o comentarios. Además, permite introducir tareas, cuestionarios, tests o material didáctico, por lo que también puede proporcionar al profesor información sobre el estado de la clase.

Google Classroom es por tanto una herramienta que incorpora las dos soluciones tecnológicas que se comentaban al principio (*backchannel* y visualización de datos), sin embargo, no dispone de ningún servicio que permita al profesor conocer en tiempo real el progreso de los alumnos en las tareas.

Al ser una aplicación relativamente reciente, está en continuo desarrollo para incorporar mejoras, pero ya cuenta con buena recepción, siendo varios los estudios realizados que concluyen un alto nivel de satisfacción por parte del alumnado y una mejoría notable en el aprendizaje y en el trabajo colaborativo [16] [17].

- **Twitter**

En la actualidad, se utiliza a menudo Twitter como herramienta *backchannel* en clases y conferencias. Los estudiantes deben comentar el tema del día con un hashtag determinado y pueden mencionar a sus compañeros o profesor mediante las menciones características de Twitter.

En [17] y [18] se analiza el uso de Twitter en casos reales de aprendizaje y en ambos estudios se concluye que el resultado ha sido positivo pero que aún existe mucho margen de mejora.

2.3.3. Problemas

El mayor problema al que se enfrenta *backchannel* es el posible aumento de la distracción en los alumnos. La creación de un hilo con una nueva conversación puede causar que los estudiantes se pierdan la explicación principal o incluso que acaben manteniendo una conversación ajena a la lección. Sobre este aspecto, en [9] se comenta que los estudiantes que se distraen con el *backchannel* probablemente estarían distraídos de cualquier otra forma.

Este problema podría verse paliado con la revisión constante del *backchannel* por parte del profesor o con la inclusión de técnicas para mantener la atención del alumno, como incluir otras tecnologías en las que el estudiante participe directamente con la explicación.

2.4. Visualización de datos

La visualización de datos es otro tipo de solución tecnológica que permite al profesor conocer el progreso de cada alumno en tiempo real y ser consciente del estado general de la clase. Mediante distintas tecnologías posibles, se captura información y se muestran los datos al profesor sobre el estado de los estudiantes.

En el caso de sesiones basadas en PBL, el profesor tendrá acceso al progreso de los alumnos en su tarea y podrá ser notificado cuando tengan una duda. En caso de sesiones basadas en una explicación principal por parte del docente, los estudiantes podrían ir contestando cuestionarios acerca de la lección impartida, proporcionando así información práctica al profesor sobre el estado de la clase.

2.4.1. Objetivo

Con este sistema, se busca que el profesor tenga una información más precisa sobre el proceso del aprendizaje, pudiendo así ofrecer una atención más enfocada a las necesidades de cada alumno. Esto también permite que el profesor pueda mejorar las explicaciones y que tenga una mayor capacidad para gestionar el ritmo de la clase.

Cuando el aprendizaje se basa en PBL, la ayuda prestada por el profesor es clave, siendo ésta en muchas ocasiones esencial para que el alumno pueda seguir trabajando. La visualización en tiempo real del progreso supone un gran apoyo para el profesor a la hora de ofrecer ayuda a los alumnos. Esto no significa que la ayuda deba ser proporcionada inmediatamente, lo cual en ocasiones se sostiene es contraproducente [19], sino que el profesor será capaz de organizar mejor la gestión de su tiempo.

Otro aspecto clave de la visualización de información es el análisis de datos a posteriori. Si los datos de progreso de cada alumno durante la clase son guardados, el profesor podrá después de la clase revisar lo sucedido. Mediante este análisis (manual o automático) se puede comprobar qué secciones dan más problemas a los estudiantes y por tanto necesitan más dedicación en la explicación. Este análisis también permitirá proporcionar un refuerzo personalizado a los alumnos con más dificultades en la materia y realizar una revisión de los materiales utilizados y el contenido perteneciente a cada sesión.

2.4.2. Aplicaciones relacionadas

La visualización de datos se puede realizar utilizando muchos tipos de tecnología, dado que hay diversas formas de mostrar el progreso de los alumnos. A continuación se describirán algunas de las más importantes:

- **Lantern**

Un ejemplo muy visual es el caso de Lantern [20], una herramienta basada en el uso de lámparas interactivas que modifican su color en función del progreso del alumno. Además, si el estudiante tiene una duda, la lámpara se iluminará de forma más intensa para llamar la atención del profesor.

En [20] se realiza un análisis completo sobre el uso de Lantern, siendo los resultados muy positivos y demostrando que se mejora considerablemente la productividad de los estudiantes y la comunicación entre éstos.

- **MTFeedback**

MTFeedback [21] es una herramienta que realiza un seguimiento en clases con tableros digitales interactivos. Cada equipo trabaja en uno de los tableros y toda la información sobre su progreso y otros datos relevantes se muestra en una tablet u ordenador que maneja el profesor. Los resultados tras el análisis realizado con esta herramienta también fueron positivos. Además, se recalca que “tanto las notificaciones positivas como las negativas fueron importantes para ayudar a los profesores a comentar el progreso con los alumnos, resolver sus dudas y animarlos a mantenerse motivados”.

- **Lumilo**

En 2018 se presenta Lumilo [22], un nuevo prototipo de gafas de realidad aumentada basado en una idea inicial presentada hace cinco años en [23] por un profesor de la Universidad Carlos III de Madrid. En estas gafas se muestra información sobre cada alumno a tiempo real, mostrando los datos que se seleccionen, su progreso o si tiene alguna duda. El prototipo se ha diseñado a partir de la experiencia de distintos profesores para optimizar su funcionamiento y la experiencia del usuario.

- **Socrative**

Socrative [24] es una de las aplicaciones más populares en este ámbito. Su funcionamiento se basa en presentar distintos tipos de cuestionarios a los alumnos que deberán ir respondiendo durante el transcurso de la explicación principal del profesor.

Se trata de un software multi-plataforma que cuenta con versión para Android, iOS y Web, con versiones separadas para profesores y alumnos. Socrative cuenta con planes premium para poder incorporar más alumnos, controlar el acceso de los estudiantes mediante un ID o crear hasta 10 clases públicas o privadas. La versión gratuita por su parte está limitada a una clase pública y 50 estudiantes por sesión.

El hecho de que pueda usarse tanto en smartphones como en tablets y ordenadores, permite aumentar su popularidad, ya que mientras el profesor está con el ordenador los alumnos pueden usar sus teléfonos, o viceversa.

La aplicación permite lanzar distintos tipos de cuestionarios a los alumnos; verdadero y falso, de respuesta corta o personalizados. Los estudiantes se conectan a una clase lanzada por el profesor y les irán apareciendo los cuestionarios propuestos en sus dispositivos. El profesor podrá ver en su pantalla las respuestas de los alumnos, mostrando los porcentajes de acierto de la clase en tiempo real.

La limitación de la aplicación es que únicamente permite introducir cuestionarios, es decir, no se pueden incorporar enunciados de ejercicios ni material didáctico. Es por tanto una herramienta enfocada a materias basadas en una metodología distinta a PBL.

2.4.3. Problemas

Aunque los resultados son muy positivos, los tres primeros ejemplos presentados requieren dispositivos específicos con un coste extra, por lo que en la mayoría de los casos no son una solución viable. Por su parte, Socrative es una herramienta orientada a sesiones con una explicación principal y no permite la visualización del progreso en materias que sigan la metodología PBL.

La visualización de la información en tiempo real además requiere un esfuerzo extra por parte del docente, porque pese a que el objetivo es facilitar su gestión, al principio requiere un cierto análisis posterior a las clases hasta hacerse con el uso de estas herramienta.

2.5. Versiones anteriores de ClassOn

En este apartado se comentaran brevemente las distintas versiones previas que existen de ClassOn, explicando el avance continuo que iban mostrando y la llegada al

punto de partida de la versión actual. Todas las versiones se desarrollan para utilizarse principalmente en la asignatura Aplicaciones Multimedia, perteneciente al tercer curso del grado universitario de Ingeniería en Sistemas Audiovisuales, teniendo siempre en cuenta la posibilidad de extrapolarlo a otras asignaturas y campos de enseñanza.

La primera versión de ClassOn se presenta en 2011 en [25]. Se trata de un prototipo con un diseño preliminar que se basa en las dos soluciones tecnológicas descritas anteriormente; *backchannel* y visualización de datos en tiempo real. En la versión desarrollada en este proyecto se ha mantenido gran parte del funcionamiento planteado en esa primera versión. En [25] se comenta que el uso de esta herramienta permite al profesor ser más consciente de lo que sucede en la clase y le permite optimizar la atención prestada a los alumnos, concluyendo por tanto que los resultados en casos reales han sido positivos.

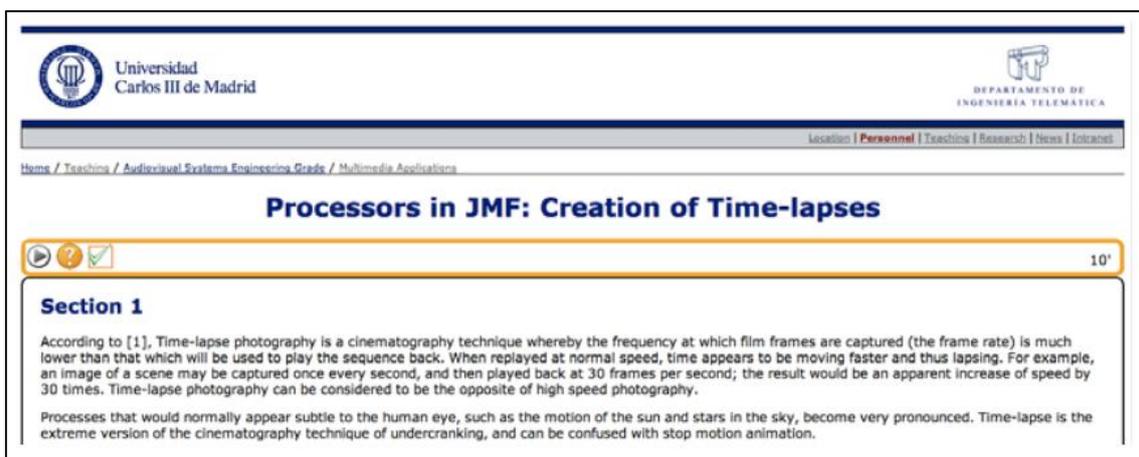


Figura 2.1. Diseño de la interfaz para alumnos de la versión de 2011 [19]

La aplicación tiene dos interfaces separadas para profesor y alumnado. En la figura 2.1 se muestra el diseño inicial para la interfaz del alumno, una apariencia sencilla con el enunciado de la sección actual, botones para indicar el progreso en dicha sección y un botón para solicitar ayuda al profesor. En la figura 2.2 se muestra la interfaz del profesor (solo se presentó una de las ventanas) en la que aparecen los puestos de cada equipo, con un muestreo visual de su progreso y un indicador del tiempo del profesor que les ha sido dedicado.

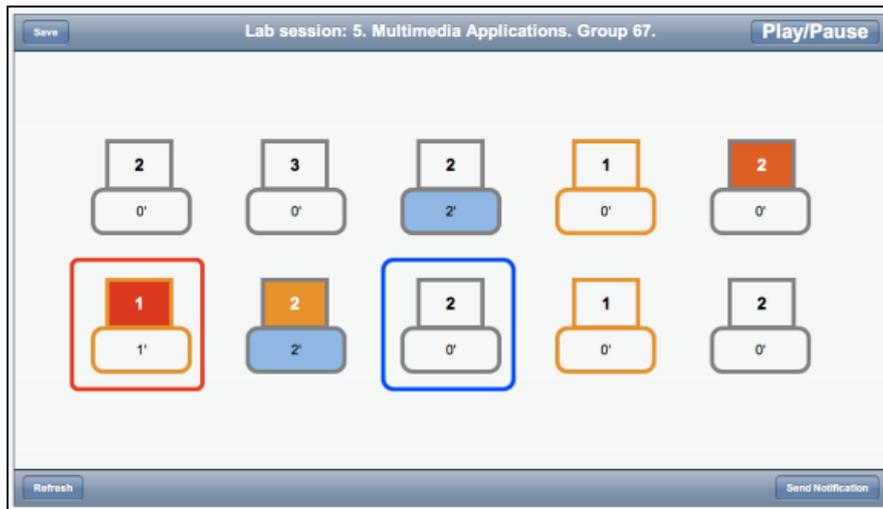


Figura 2.2. Diseño de la interfaz para el profesor de la versión de 2011 [19]

Un año más tarde, basado en las ideas de [25] se desarrolla un prototipo con un diseño completo y nuevas funcionalidades. Esta versión se describe en [26] y al igual que la anterior, basa su diseño en dos interfaces separadas, siendo la de los alumnos una versión limitada para ordenadores y la del profesor una versión limitada para tablets. El diseño se mantiene muy similar a la versión anterior pero con la introducción de algunos cambios sugeridos por profesores y alumnos.

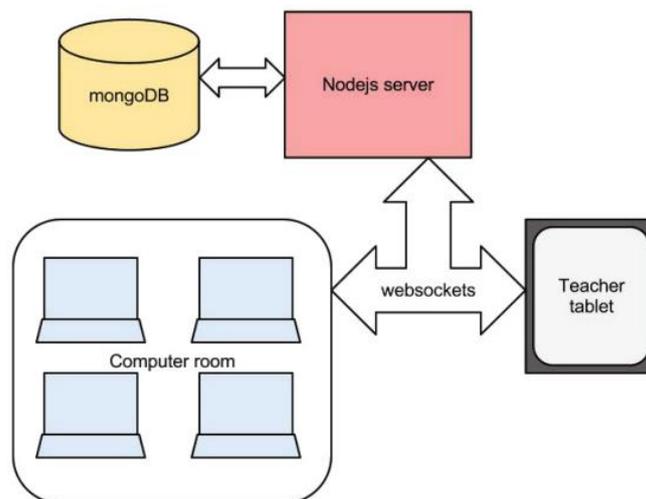


Figura 2.3. Arquitectura de las tecnologías utilizadas en la versión de 2012 [20]

La arquitectura de la aplicación se muestra en la figura 2.3, en la que se indica las distintas tecnologías utilizadas; mongoDB como base de datos y Node.js como servidor.

En [26] también se evalúa el funcionamiento de la aplicación, realizando un estudio llevado a cabo durante cinco sesiones en cuatro clases con grupos de 20 a 30 alumnos. Los profesores que la utilizaron indicaron como puntos positivos que les ayudaba a conocer al momento el progreso de cada estudiante y que gracias a la aplicación hacían un reparto más justo de la atención dedicada a cada alumno. Como puntos negativos subrayaron que no solían leer las dudas en la aplicación, sino que se las consultaban directamente a los estudiantes, así como la necesidad de mejorar la interfaz de usuario.

Los alumnos por su parte valoraron positivamente la aplicación, reseñando la necesidad de mejorar la interfaz del estudiante y cuestionando la necesidad de añadir las dudas por escrito en la aplicación, prefiriendo éstos hacer las consultas directamente en persona.

En [27] se comenta la posibilidad de utilizar la aplicación en clases con un número cuantioso de alumnos. Se concluye que, modificando algunas características, ClassOn podría llegar a ser una herramienta muy útil para grupos muy grandes, siendo fundamental la introducción de un sistema de ayuda colaborativa entre estudiantes y la necesidad de adaptar las interfaces para este nuevo escenario (ejemplo en la figura 2.4).

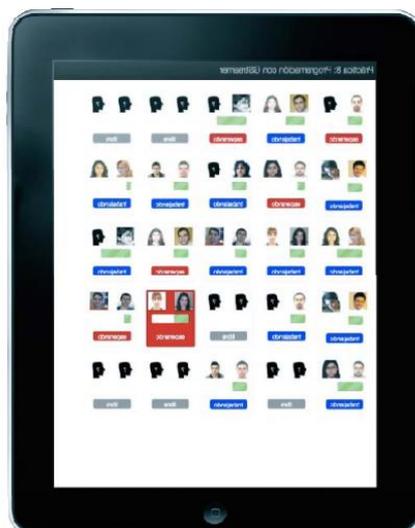


Figura 2.4. Interfaz del profesor en tablets para un escenario con muchos alumnos [21]

2.6. Estado del arte de la tecnología utilizada

En la aplicación se utilizan distintas tecnologías para su desarrollo. A continuación se describirá brevemente cada una, indicando su función en la aplicación.

- **HTML5**: lenguaje estándar de marcado para la creación de páginas web, instaurado por el consorcio W3C [28]. Es fundamental en todas las páginas web y es el lenguaje que han adoptado todos los lenguajes web.
- **CSS**: lenguaje que describe el estilo de un documento HTML. Define como deben mostrarse los elementos HTML. Presente en todas las páginas web y mantenido también por el consorcio W3C.
- **LESS** [29]: preprocesador de hojas de estilo que se compila a CSS. Se basa en SASS, otro preprocesador de CSS y está escrito en JavaScript. Less permite la creación de variables, el anidamiento de clases y la creación de mixins y funciones. Actualmente está muy extendido en el desarrollo web, ya que facilita el trabajo del desarrollo y permite trabajar con un código más limpio.
- **JavaScript**: lenguaje de programación basado en el estándar ECMAScript. Uno de los lenguajes principales en el desarrollo web, utilizado principalmente en el lado del cliente. La función de JavaScript en una página web es permitir ejecutar código en el propio navegador cliente, es decir, se encarga de gestionar el comportamiento de la web y la funcionalidad de los componentes. En ClassOn se utiliza para el desarrollo del lado del cliente o front-end.
- **Python**: lenguaje de programación para desarrollo genérico. Es uno de los lenguajes más usados en distintas tecnologías por su sencillez y limpieza de código. En ClassOn se utiliza en el lado del servidor o back-end.
- **Flask** [30]: micro-framework escrito en Python para el desarrollo web. Se denomina micro-framework porque no requiere ninguna librería adicional. Su uso

está muy extendido ya que optimiza el desarrollo web utilizando el lenguaje Python y dispone de numerosas extensiones.

En ClassOn se utiliza en el back-end y se añaden varias extensiones:

- Flask-SocketIO: permite la comunicación entre el lado del cliente y el lado del servidor haciendo uso de Flask.
 - Flask-MySQLdb: permite trabajar con bases de datos MySQL haciendo uso de Flask.
-
- **MySQL** [31]: sistema de gestión de bases de datos basado en el modelo relacional. Es uno de los sistemas más extendidos en el desarrollo web y se utiliza para alojar todos los datos necesarios en ClassOn.
 - **Git** [32]: sistema de alojamiento y control de versiones para aplicaciones. Se utiliza GitHub como repositorio principal para todo el código.

3. DESCRIPCIÓN FUNCIONAL

3.1. Introducción

En este capítulo se realizará una descripción funcional del sistema y se describirá la estructura de la aplicación. Además, se comentarán las soluciones tecnológicas utilizadas (*backchannel* y visualización de datos) y su aplicación dentro de ClassOn.

3.2. Objetivo de la aplicación

El objetivo de ClassOn es dar soporte a la interacción profesor-alumno y proporcionar al docente información del progreso de los alumnos durante la sesión. Para conseguir esto, se utilizan dos de las soluciones tecnológicas descritas previamente; *backchannel* y visualización de datos en tiempo real. ClassOn pretende, mediante el uso de estas dos soluciones, mejorar las aplicaciones ya existentes en el mercado, ya que todas sufren la carencia de algunas características necesarias y no aprovechan del todo las ventajas que pueden ofrecer estos sistemas.

ClassOn se diseña para utilizarse en clases informáticas donde todos los alumnos tienen acceso a un ordenador con conexión a internet. Es una aplicación orientada para asignaturas basadas en PBL, ya que se diseña para que el alumno trabaje de forma autónoma a través de las secciones habilitadas por el profesor. No obstante, puede ser útil también para otro tipo de metodologías, siempre que el profesor emplee las secciones correctamente y los alumnos hagan un uso adecuado de la aplicación.

Para la correcta aplicación de estas soluciones, se busca que la aplicación sea intuitiva, sencilla y eficiente técnicamente. Intuitiva porque es esencial que los alumnos sepan manejar con la aplicación desde el principio, de forma que la entiendan como una herramienta práctica y esto mejore su atención y su aprendizaje. Si la aplicación es complicada de manejar o requiere de un aprendizaje previo, es probable que los alumnos muestren una pérdida de interés y no aprovechen los recursos que se les presentan con ClassOn.

Sencilla porque el profesor debe ser capaz de tener toda la información al instante, evitando complejidades técnicas o sobresaturación por detalles. Es fundamental que el profesor tenga acceso a la información que requiere, un exceso de datos en la pantalla puede terminar confundiendo al usuario. Por último, eficiente porque debe funcionar de forma rápida y sin errores, algo esencial en cualquier herramienta web.

Además, se requiere que la aplicación sea lo más transparente y menos intrusiva posible, haciendo un uso responsable de los datos y solicitando su consentimiento al usuario.

3.3. Estructura básica

ClassOn es una aplicación única para profesores y alumnos, es decir, no existen versiones diferentes para cada uno, sino que dentro de la propia aplicación existe un bloque para el alumno y un bloque para el profesor con distinta interfaz y funcionalidad. Con esto se busca simplificar la tecnología y unificar lo más posible el formato, además de agilizar el proceso de comunicación de los datos y evitar errores.

Pese a que es una única versión en una misma dirección web, la aplicación se divide dentro de si misma en dos bloques. Primero, los usuarios inician sesión y en función de su tipo de cuenta se les redireccionará a un bloque o a otro. Para iniciar sesión, los usuarios deben previamente registrarse, proporcionando su nombre, correo y una foto tipo DNI.

En principio, cualquiera puede registrarse como alumno o como profesor, pero esto se podría limitar para evitar posibles accesos fraudulentos.

3.3.1. Bloque del profesor

Una vez el profesor ha iniciado sesión, éste será redirigido a una pantalla con un panel de opciones. En dicho panel podrá seleccionar entre; crear una clase nueva, abrir una clase ya existente o crear secciones.

En la opción “Crear una clase nueva” se permitirá al profesor crear un mapa para un aula nueva. Mediante un formulario, seleccionará el número de filas y columnas de la clase y su nombre. Además, podrá comprobar en dicha pantalla la distribución de los sitios mediante un boceto del mapa. Esta clase se almacena en la base de datos, por lo que solo es necesario crearla una vez y podrá ser seleccionada por todos los profesores registrados. Esto se diseña así porque las aulas informáticas suelen usarse para distintas materias impartidas por distintos profesores y si varios utilizan la aplicación, lo idóneo es que no deban crear un aula que ya se creó con anterioridad. Además, esto permite reutilizar el mapa para distintas ediciones de un curso.

La segunda opción del panel es “Abrir clase ya existente”, la cual dirigirá al usuario a una nueva pantalla en la que deberá seleccionar una clase de las existentes y una tarea de las creadas en la tercera opción. Tras esta selección, se abrirá el mapa de dicha clase con dicha tarea asignada y pasará a estar disponible para los alumnos que utilicen la aplicación. Si la asignatura no lo requiere, el profesor podrá crear una tarea vacía de forma que solamente se aproveche la funcionalidad de la gestión de dudas pero no la de visualización de información.

La tercera opción es “Crear nueva tarea”, que permitirá al usuario añadir contenidos y organizarlos dentro de secciones. Es importante que el profesor distribuya correctamente los contenidos por secciones en la tarea, ya que esto le permitirá observar su progreso correctamente en el mapa de la clase. Esto se refiere a que si se añade todo el contenido en una sección, todos los alumnos estarán permanentemente en dicha sección y no podrá distinguir su progreso. Todos los contenidos de las tareas se almacenarán en la base de datos, de modo que los profesores puedan reutilizar las tareas de años anteriores.

En la pantalla de “Crear nueva tarea” se utilizara un editor de texto que permite crear ahí las secciones o bien añadir directamente la versión HTML de la lección. Esto se diseña así con vistas a mejorar el rendimiento de la aplicación en asignaturas de programación web, para que se permita introducir componentes directamente y mejorar así el aprendizaje de los estudiantes. Gracias a esta opción de inserción HTML además se

permite a los profesores añadir directamente contenido de blogs o sitios web que puedan haber creado con anterioridad.

Una vez abierta una clase, el profesor siempre podrá volver al panel inicial, lo que le permitirá abrir una nueva clase o crear una nueva tarea en cualquier momento.

3.3.2. Bloque del alumno

El diseño del alumno es más sencillo, ya que no requiere ninguna herramienta para la gestión de las clases, sino que es un usuario que va avanzando por distintas pantallas basadas en la creación de la tarea hecha por el profesor.

En el tablero del alumno, aparecerán todas las clases lanzadas disponibles. Cada clase aparece representada con un nombre y una tarea. Aunque es un caso remoto, puede llegar a haber dos clases simultáneas en un aula informática, por lo que se añade la tarea asignada para poder distinguirlas. Una vez el alumno ha accedido a su clase correspondiente, le aparecerá una nueva pantalla con el mapa del aula. El usuario entonces selecciona su sitio en el aula y éste le será asignado, apareciendo así en el mapa del profesor.

El alumno entonces accede a la tarea asignada por el profesor y puede ir navegando por las secciones mientras avanza o volver atrás si así lo requiere. En cada sección, tendrá un formulario para subir dudas, que quedarán anexadas a dicho sección. Junto al formulario aparecerá un tablón con todas las dudas respecto a dicho capítulo para que los alumnos puedan ver y contestar las dudas de sus compañeros.

En todas las pantallas, en la barra de herramientas superior aparecerá la opción “Añadir compañero”. Mediante esta opción, se accede a un formulario para que otro usuario inicie sesión. La sesión es conjunta, ya que se entiende que ambos usuarios están en el mismo puesto y, por tanto, trabajan colaborativamente. Ambos usuarios aparecerán juntos en el mismo sitio del mapa y su progreso por la sección será compartido. Se diseña de esta

manera, porque es habitual que en una clase haya puestos con una persona y puestos con dos, pero todos deben iniciar sesión para que el profesor sepa quien está en clase durante la sesión. Cuando el usuario cierre sesión o salga de la aplicación, automáticamente se saldrán todos los alumnos de ese puesto. Esta es una limitación debida a la gestión de los usuarios realizada por el back-end.

3.4. Backchannel en ClassOn

Se decide utilizar la tecnología *backchannel* en la aplicación para que la comunicación profesor-alumno sea eficaz y fluida, así como para promover la colaboración entre compañeros. Como se explicaba en el capítulo 2, la creación de una conversación a tiempo real ayuda a mantener la atención de los alumnos y les permite resolver dudas al instante, ayudados por el profesor o por sus compañeros.

En el caso de ClassOn, al estar enfocado para clases basadas en PBL, no existe una explicación que se extienda durante la clase, por lo que no tiene sentido anclar dudas o comentarios a diapositivas. Lo fundamental para PBL es mejorar y agilizar la resolución de las dudas. Para esto, se diseña un sistema que permita a los alumnos publicar sus dudas en cada sección, pudiendo añadir toda la información que consideren necesaria para su resolución.

El funcionamiento de la gestión de dudas se representa en la figura 3.1. Primero, el alumno publica su duda en la aplicación y en ese momento se empieza a contabilizar el tiempo de espera. Tras esto, la duda se publica en un muro presente en la aplicación y ahí todos los alumnos y el profesor tienen acceso a ella y pueden ir a ayudar al alumno a su puesto o directamente contestarla mediante la aplicación. Si el alumno considera que su duda ha sido resuelta, la marcará como cerrada y desaparecerá del muro de dudas.

Si la duda permanece sin resolver, el alumno deberá esperar a que el profesor le atienda. Esta fase suele ser la más caótica en las clases. Algunos alumnos tardan demasiado en ser atendidos respecto a sus compañeros, otros son atendidos durante un exceso de tiempo y

otros se quedan en la primera parte del proyecto por falta de atención. Además, los alumnos suelen dejar la tarea a un lado para volcarse en captar la atención del profesor, lo cual les impide seguir avanzando o tratar de resolver la duda por su cuenta.

Para solucionar esto se diseña un algoritmo que actúa de gestor de dudas. Este gestor de dudas recomendará al profesor que orden debe seguir al resolver las dudas, considerándose el tiempo de espera del alumno, su progreso y el tiempo que ya ha sido atendido por el profesor durante la clase.

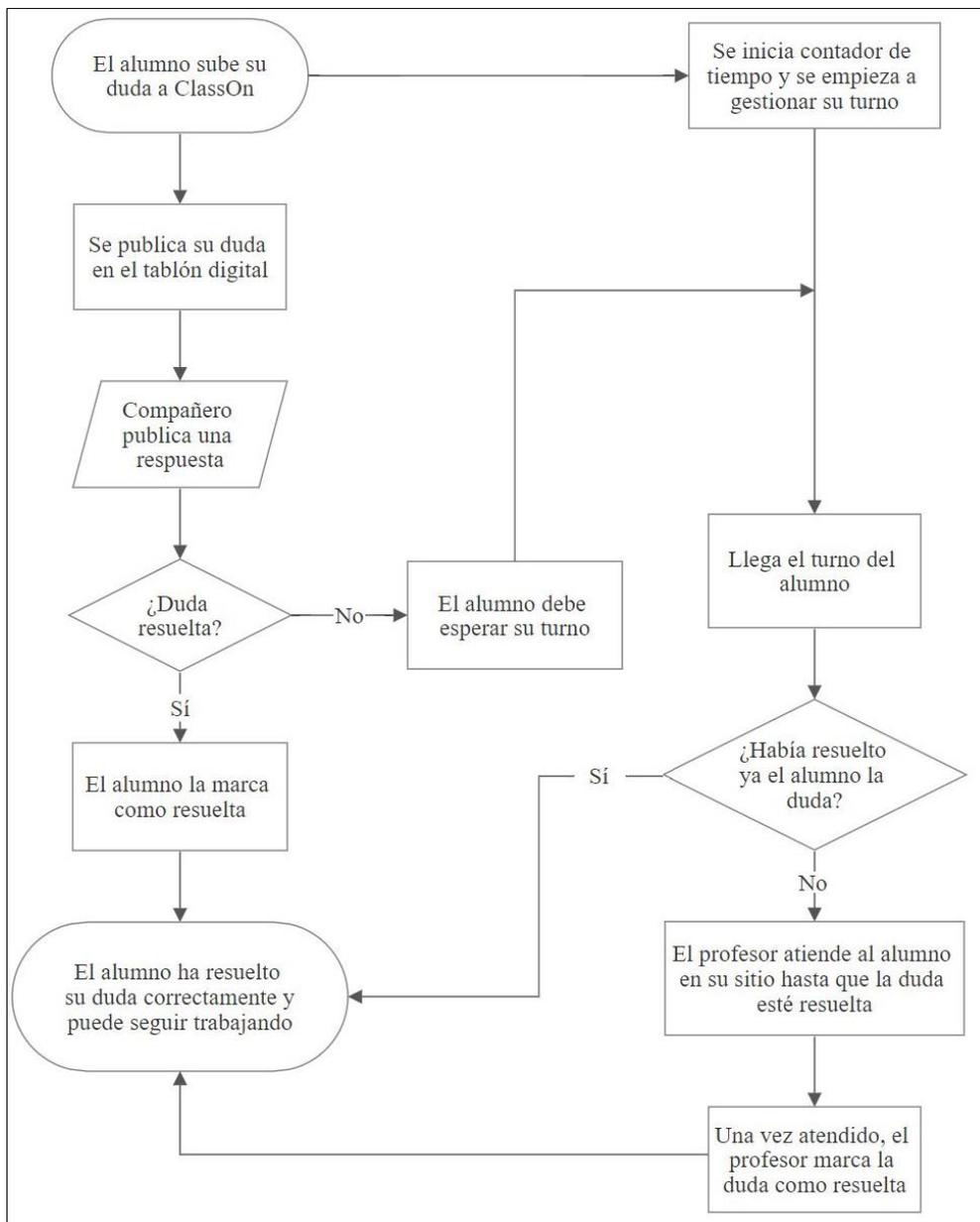


Figura 3.1. Diagrama de flujo de la gestión de dudas

Es importante promover la ayuda entre los propios alumnos, ya que esto agilizará considerablemente la gestión de las dudas y mejorará su aptitud de trabajo en equipo, algo esencial en las metodologías PBL.

Se prescinde de la creación de un chat dentro de la aplicación porque se considera más práctico el uso de un tablón de dudas. Aunque es cierto el chat que permite crear una conversación más fluida, se considera que esta tecnología fomenta la distracción del alumnado y no beneficia el funcionamiento de las clases.

3.5. Visualización de datos en ClassOn

La visualización de la información de los alumnos es uno de los aspectos fundamentales de ClassOn. Es una herramienta clave para optimizar el funcionamiento de la clase, ya que gracias a esto el profesor identifica rápidamente el progreso de los estudiantes, puede ver si todos se bloquean en una misma sección y puede ayudar a los que lleven un ritmo más lento. El muestreo visual de la información desarrollada en ClassOn se basa en el diseño de Lantern [20] pero se añade un indicador numérico con el número de la sección en la que está cada alumno o equipo de alumnos.

La pantalla del mapa de la clase está desarrollada para optimizar este seguimiento, de forma que la información sea completamente visual y accesible. Se diseña para que todos los puestos aparezcan siempre en la pantalla y así el profesor sea capaz de identificar el progreso de cada uno rápidamente.

Esta visualización de datos a tiempo real no está desarrollada en ninguna aplicación similar y es una de las grandes ventajas de ClassOn. En versiones futuras se podrían utilizar los datos del seguimiento (mediante *learning analytics* [33]) para su posterior análisis con el fin de mejorar las clases, reforzar el contenido en los capítulos que más tiempo bloqueen a los alumnos y evitar que haya estudiantes que se quedan atrás en la materia.

3.6. Consideraciones técnicas previas

ClassOn es una aplicación web pensada para cualquier navegador (Chrome, Mozilla, Opera...) utilizado tanto en ordenadores como en tablets. Esto se diseña así porque lo óptimo es que el profesor utilice la tablet mientras se desplaza por los distintos puestos de la clase, de esta forma el acceso a la información es inmediato y más ágil.

El diseño del bloque de los alumnos también está pensado para ser adaptativo, ya que puede haber estudiantes que prefieran utilizar una tablet personal para el uso de ClassOn ante la ausencia de ordenadores suficientes para todos e incluso puede haber entornos académicos en los que todos los alumnos trabajen con tablets.

Se requiere por tanto que la aplicación tenga un diseño adaptativo para anchos comprendidos entre 1024px (tablet en horizontal) y 1920px. No se contempla el diseño para dispositivos móviles porque se considera que el tamaño de la pantalla no es suficiente para poder utilizar la aplicación correctamente.

4. IMPLEMENTACIÓN DE LA SOLUCIÓN

4.1 Introducción

En este capítulo se describirá el sistema desarrollado. Se definirá primero a alto nivel, analizando la arquitectura tecnológica y especificando el proceso seguido para el back-end (lado del servidor) y para el front-end (lado del cliente).

Tras esto, se analizarán las distintas pantallas de la aplicación, comentando su diseño y especificando sus distintas funcionalidades.

4.2. Arquitectura tecnológica

La aplicación está implementada de acuerdo a la arquitectura cliente-servidor y se puede dividir en dos grandes bloques; front-end y back-end. El back-end engloba el lado del servidor y su comunicación con el lado del cliente y la base de datos, mientras que el front-end gestiona la interfaz que ve el usuario en su dispositivo y añade funcionalidad a los componentes. En la figura 4.1 se muestra el esquema de la arquitectura tecnológica de ClassOn:

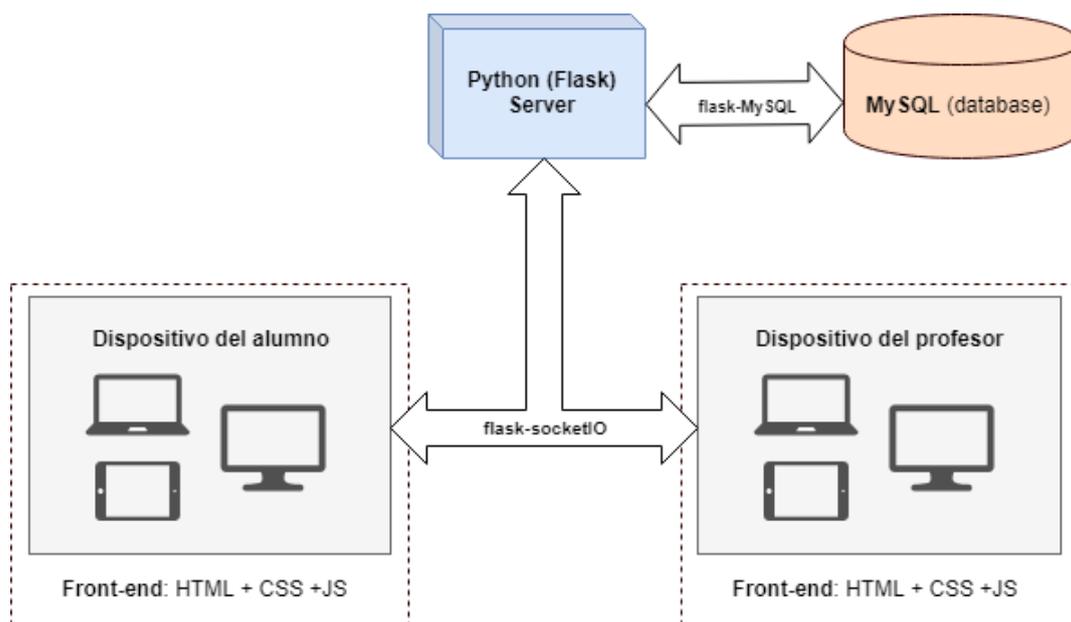


Figura 4.1. Esquema de la arquitectura tecnológica de ClassOn

4.2.1 Back-end

El back-end es la parte de la aplicación desarrollada para la gestión del lado del servidor y su comunicación con el front-end. Se encarga de la gestión de los datos y su procesamiento, de la transmisión de paquetes en toda la aplicación y del almacenamiento en bases de datos.

El desarrollo del back-end de ClassOn fue programado por Jorge Frías, un antiguo estudiante de la Universidad Carlos III de Madrid que desarrolló el lado del servidor y preparó un prototipo de la aplicación que es a partir del cual se ha terminado de desarrollar esta versión de ClassOn.

Sobre el trabajo de Jorge en el back-end se han incluido algunas funcionalidades nuevas, tanto en la gestión de datos como en el almacenamiento en la base de datos. Además, se han ido arreglando fallos sobre su código que se han detectado durante el desarrollo de esta versión. Estas son todas las modificaciones que se han realizado sobre el back-end en este proyecto:

- Corrección de errores en la comunicación. Se perdía información de la sesión y no se mandaban correctamente los datos del progreso y las dudas de los alumnos.
- Creación de una nueva herramienta en el panel del profesor que permite abrir una clase ya existente.
- Uso de imágenes de perfil de una biblioteca para los alumnos. Esto ha requerido también modificar los datos que se enviaban a través del formulario.
- Almacenamiento de las tareas creadas como código HTML para su posterior representación manteniendo el formato con el que se crea en el formulario.
- Corrección de pequeños errores.

El desarrollo está realizado en el lenguaje Python, uno de los principales lenguajes para el lado del servidor. El almacenamiento de datos se realiza utilizando una base de datos MySQL, donde se guardan todos los datos que van siendo creados en la aplicación. Además, se utilizan los siguientes **frameworks**:

- **Flask**

Flask [30] es fundamental en el desarrollo del lado del servidor de ClassOn. Se trata de un framework para Python creado para facilitar y mejorar el desarrollo web de una aplicación. Incluye algunas extensiones creadas para aplicaciones Flask:

- **Flask-SocketIO** [34]: utilizado para el desarrollo de la comunicación entre el lado del cliente y el lado del servidor. Es un framework creado para poder utilizar SocketIO fácilmente en aplicaciones que utilizan Flask.
- **Flask-MySQL** [35]: permite la gestión de base de datos MySQL en aplicaciones que utilizan Flask en el lado del servidor.
- **Flask-WTF** [36]: permite la integración de WTForms en aplicaciones Flask. WTForms es un framework para Python que permite la insertar formularios en HTML y obtener los datos de respuesta directamente en el lado del servidor para poder almacenarlos en la base de datos. Son formularios modificables por lo que no tiene limitaciones en cuanto a diseño o funcionalidad.

4.2.2. Front-end

El front-end es la parte del desarrollo de la aplicación que se encarga de generar la interfaz gráfica con la que el usuario interactúa. A través de los lenguajes para el desarrollo web; HTML y CSS, se desarrolla la apariencia digital que hace uso de los datos generados con el back-end de la aplicación.

La gestión del funcionamiento de las páginas se hace mediante JavaScript. En la versión inicial diseñada por Jorge Frías se incluía un prototipo funcional, por lo que parte de ese prototipo se ha utilizado como base para el front-end desarrollado finalmente en este proyecto.

Para el desarrollo del front-end se utilizan los siguientes **frameworks y componentes de terceros**:

- **LESS**

Se utiliza Less para realizar el desarrollo de la hoja de estilo para toda la aplicación. Mediante un software se realiza la compilación para transformar el archivo Less a un archivo CSS. En este caso, se utiliza WinLess [37], un software específico para Windows.

- **Key_animations.css**

Se utiliza esta librería open-source de animaciones para aplicarlas en los elementos de la aplicación. Se importan a la hoja de estilos mediante una importación con Less.

- **Bootstrap 4.0**

Se utilizan varios componentes de la librería Bootstrap [38] en la aplicación. Ayuda a mejorar la apariencia de la aplicación y facilita el desarrollo de distintas partes de la página. Se añade directamente en el Layout principal para que todas las páginas puedan hacer uso de esta librería.

- **Summernote**

Summernote [39] es un editor de texto insertable en HTML basado en Bootstrap. Se utiliza por su capacidad para añadir formato al texto y su posibilidad de importar y exportar código HTML, algo fundamental para la creación de las tareas de ClassOn que se detallará más adelante.

- **Icons8**

Todos los iconos utilizados son de la librería de iconos de Icons8 [40] y se pueden usar gratuitamente a cambio de incluir un link a la web en la aplicación.

4.2.3. Estructura del código

Para el desarrollo de la aplicación, el código se distribuye de la siguiente forma:

- **HTML:** La estructura se basa en la distribución inicial por carpetas implementada en el back-end. Dentro de cada carpeta categórica, existe una carpeta *Templates* donde están los archivos HTML que contienen el desarrollo de cada una de las pantallas.
- **CSS:** Existe una única hoja de estilo para toda la aplicación cuyo desarrollo está hecho en un archivo Less. Este archivo se compila a un archivo CSS que es el que se añade a HTML en el layout principal, de forma que todas las pantallas de la aplicación lo incluyen y comparten la misma hoja de estilos.
- **Python / JavaScript:** Cada carpeta contiene sus correspondientes archivos Python y JavaScript, encargados de controlar tanto el back-end como el front-end. Las carpetas se organizan en cinco módulos; ‘módulo de estudiante’, ‘módulo de profesor’, ‘módulo *static* (servidor)’, ‘modulo *home*’ y ‘módulo de tareas’.

4.3. Descripción de las pantallas y sus funcionalidades

4.3.1. Componentes genéricos

Existen componentes genéricos que se repiten en todas las pantallas y que están definidos en un layout HTML principal en el que luego se van introduciendo los demás. En este layout genérico además se inserta la hoja de estilos CSS para que todas las pantallas hagan uso de este archivo.

El componente que se introduce en este layout es la *Action Bar* o barra de navegación. Una barra superior que se repite en todas las pantallas y que permite al usuario desplazarse a través de la aplicación y utilizar otras herramientas. En la figura 4.2 se observa una muestra de esta barra:



Figura 4.2. Barra de navegación

El icono de ClassOn permite al usuario volver a la pantalla inicial y es permanente en todas las pantallas. Los elementos de la derecha irán variando según la pantalla y el tipo de usuario (profesor o alumno) y disponen de una animación para hacer entender al usuario que son herramientas de navegación.

Otro elemento común en todas las pantallas es el título y el icono de la pestaña. Pese a que en cada pantalla la URL va modificándose, estos elementos se mantienen fijos. En la figura 4.3 se representa el resultado en el navegador Chrome.



Figura 4.3. Diseño de la pestaña en el navegador Chrome

4.3.2. Pantalla inicial

Al entrar en la aplicación, lo primero que ve el usuario es la pantalla inicial o *Home*. Esta pantalla funciona como portada de la aplicación y debe tener una apariencia simple para que el usuario sepa manejarse con ella desde el principio. En la figura 4.4 se muestra esta pantalla para un dispositivo con un ancho de pantalla de 1366 píxeles.

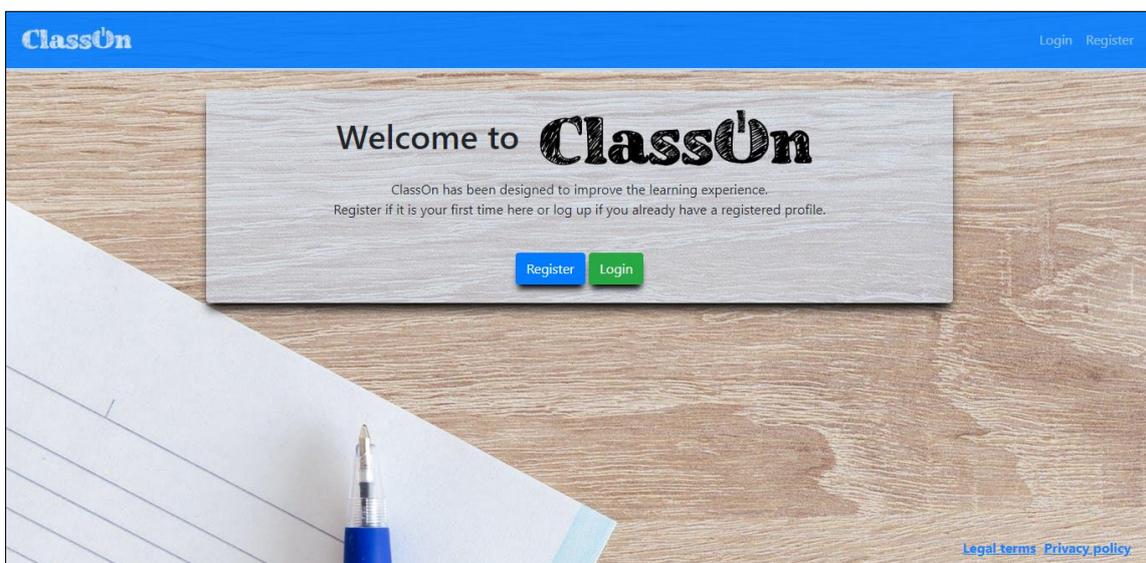


Figura 4.4. Pantalla inicial de la aplicación

Esta pantalla se adapta a distintos anchos, ya que como se comentaba en el capítulo de diseño, todas las pantallas deben ser válidas para navegadores en tablets y en distintos tamaños de ordenador. Los elementos de esta pantalla disponen de una animación *hover* al pasar el ratón por encima de ellos para hacer ver al usuario que son elementos clickeables y mejorar así su experiencia visual en la aplicación.

Como es la pantalla inicial a la que accede el usuario, se muestran las dos opciones iniciales que puede realizar en dos botones llamativos en el centro de la pantalla. Si el usuario vuelve a esta pantalla habiendo entrado ya con su usuario en la aplicación, estos botones desaparecen, y el elemento contenedor *ClassOn* se convierte en un elemento clickeable para el usuario, que le mandará a la pantalla del panel de herramientas o *dashboard*.

Además, en la parte inferior derecha de la pantalla, aparecen dos enlaces, uno para los términos legales y otro para la política de privacidad. Estos elementos son necesarios tal y como se detalla más adelante en el capítulo 6 relativo al marco regulador.

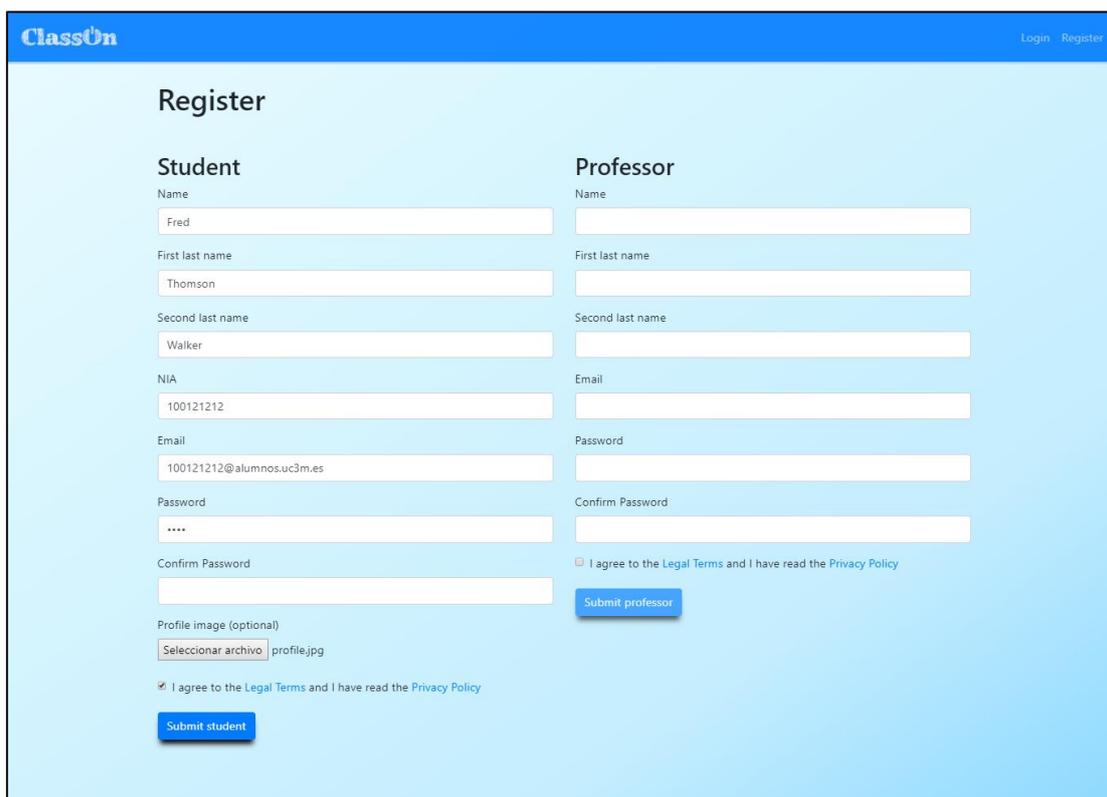
4.3.3. Registro

Tras la pantalla inicial, el usuario hará click en una de las dos opciones; registrarse o entrar con su usuario. Cada una de las opciones tiene una pantalla con un formulario, aunque la apariencia de ambas es muy similar.

Si es la primera vez que entra, el usuario procederá a registrarse y accederá a la pantalla mostrada en la figura 4.5 ¹, tanto si es alumno como si es profesor. En esta pantalla, el usuario se registrará en su rol correspondiente rellenando un formulario u otro. Al estudiante se le solicita un campo obligatorio más, su NIA (numero de identificación de alumno) y un campo opcional, una imagen de perfil.

¹ En la figura 4.5 se hace la captura de pantalla con una resolución diferente para mostrar la página completa

El botón para registrarse está deshabilitado hasta que el usuario marca la casilla mostrando que está de acuerdo con los términos y condiciones. Una vez habilitado, si el formulario está correctamente cumplimentado, se guardan todos los datos en la base de datos y el usuario queda registrado en ClassOn. Si hay un error en el formulario (un campo sin rellenar o las contraseñas no coinciden), se le indica al usuario que campo debe revisar.



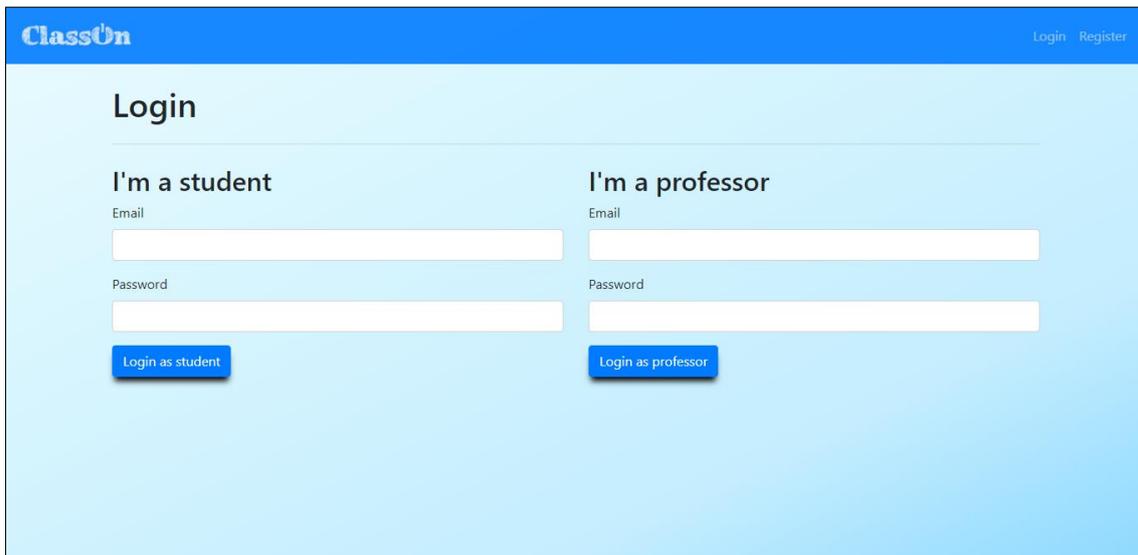
The screenshot shows the 'ClassOn Register' page. It features a blue header with the 'ClassOn' logo and 'Login Register' links. The main content is divided into two columns: 'Student' and 'Professor'. The 'Student' column includes fields for Name (Fred), First last name (Thomson), Second last name (Walker), NIA (100121212), Email (100121212@alumnos.uc3m.es), Password (masked with dots), and Confirm Password. It also has a 'Profile image (optional)' section with a file selection button and a checkbox for terms and conditions. A 'Submit student' button is at the bottom. The 'Professor' column includes fields for Name, First last name, Second last name, Email, Password, and Confirm Password. It also has a checkbox for terms and conditions and a 'Submit professor' button.

Figura 4.5. Pantalla de registro para estudiantes y profesores

La imagen de perfil se mantiene como opcional para evitar que un usuario no pueda registrarse por no disponer de una foto en dicho ordenador. Se le asignará a cada usuario sin imagen de perfil una imagen aleatoria de una biblioteca de iconos almacenada en el servidor. Actualmente esta imagen no puede guardarse en el servidor mediante la base de datos, por lo que para asignar esta imagen al usuario sería conveniente introducirla en la carpeta del servidor manualmente.

4.3.4. Inicio de sesión

Si el usuario ya está registrado, en la pantalla inicial seleccionará la opción de iniciar sesión. La pantalla que aparece entonces es más simple que la de registro, ya que aparece un formulario con solo dos campos como se muestra en la figura 4.6.



The screenshot shows the ClassOn login interface. At the top, there is a blue navigation bar with the 'ClassOn' logo on the left and 'Login Register' on the right. The main content area has a light blue background and is titled 'Login'. Below the title, there are two columns. The left column is for 'I'm a student' and the right column is for 'I'm a professor'. Each column contains an 'Email' input field, a 'Password' input field, and a blue button labeled 'Login as student' or 'Login as professor' respectively.

Figura 4.6. Pantalla de inicio de sesión para estudiantes y profesores

El usuario aquí introducirá sus datos de acceso dentro del formulario correspondiente a su rol. Tras rellenar los campos de email y contraseña y hacer click en el botón de iniciar sesión, si los datos coinciden con un usuario almacenado en la base de datos, entonces el usuario accede a la aplicación. Si por el contrario los datos no se corresponden con ningún usuario registrado, aparecerá un mensaje de error indicándole al usuario que lo intente de nuevo.

Es importante que los usuarios recuerden la contraseña, ya que esta versión no incluye ninguna opción para recuperarla en caso de que el usuario se olvide.

4.3.5 Panel del profesor

Si el usuario inicia sesión correctamente como profesor le aparecerá una pantalla nueva; el panel del profesor o *dashboard*. En esta pantalla se muestran tres botones, cada uno con una herramienta que el profesor puede utilizar y que le mandará a una pantalla nueva. Estas herramientas son “Crear una clase nueva”, “Abrir una clase existente” y “Crear una tarea”. En la figura 4.7 se muestra la pantalla que contiene el panel del profesor.

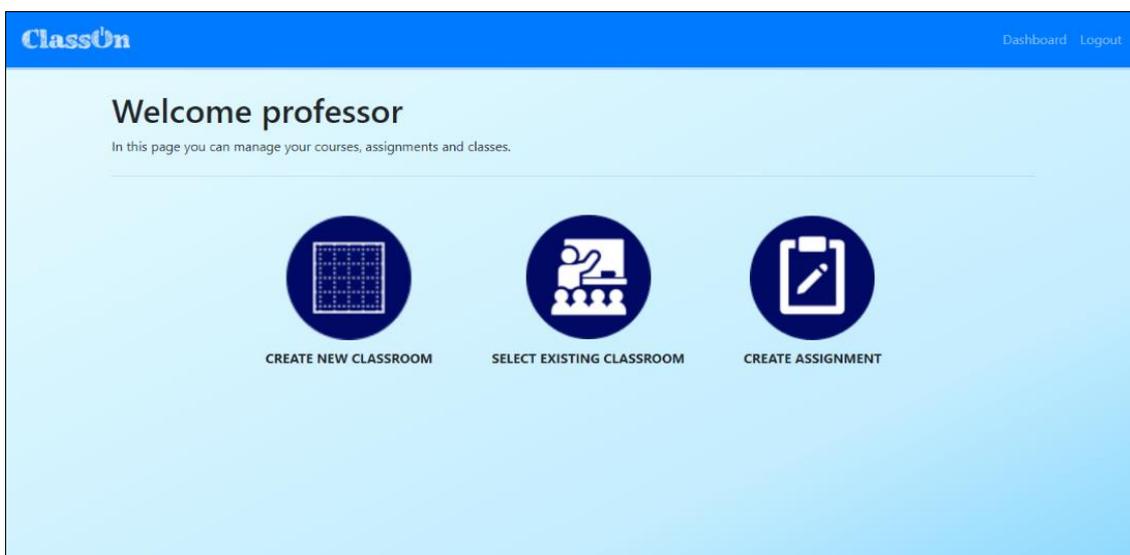


Figura 4.7. Panel del profesor

Se observa además que en la barra superior ahora aparecen dos opciones nuevas, una para volver a este panel desde cualquier pantalla y otra para salir de la sesión del usuario.

4.3.6 Creación / selección de una clase

Las dos primeras herramientas del panel del profesor son para lanzar una clase con la aplicación y utilizar sus funcionalidades de backchannel y la visualización de datos en tiempo real. Con la primera opción, el profesor crea una clase desde cero mediante un formulario tal y como se representa en la figura 4.8.

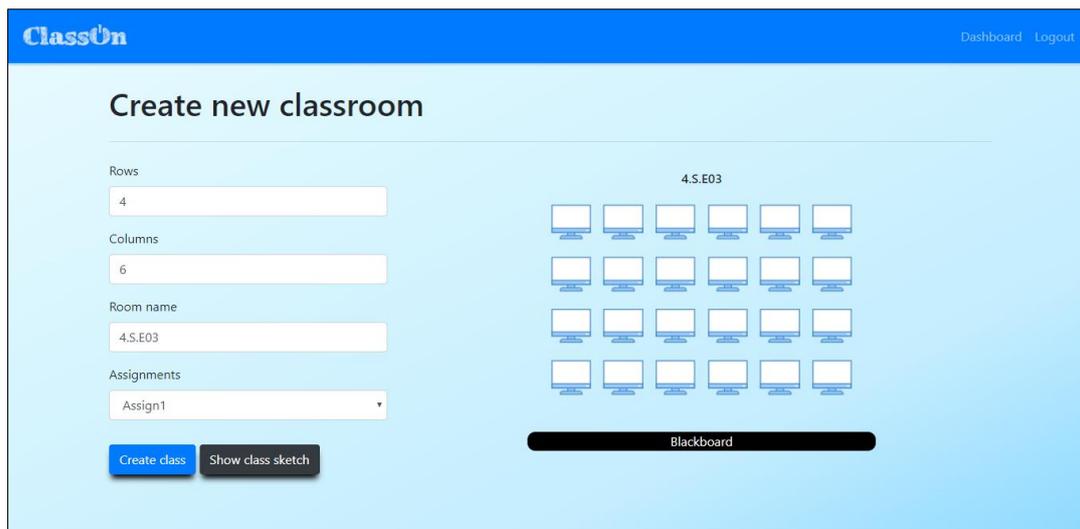


Figura 4.8. Pantalla de creación de una clase nueva

El usuario introduce el número de filas y columnas, el nombre de la clase y la tarea que le quiere asignar para esa sesión. Mediante el botón *show class sketch* se genera un boceto de esa clase, de forma que el usuario puede comprobar si está introduciendo correctamente los datos requeridos. El diseño de este boceto es adaptativo a distintos anchos de pantalla, por lo que se genera correctamente también en tablets y en ordenadores con una pantalla más grande.

Se le asigna una tarea a dicha clase porque, tras su creación, la clase es directamente lanzada y requiere un contenido para su uso². En la base de datos, sin embargo, no se asignará dicha tarea a la clase, ya que seguramente querremos asignar otras tareas en las siguientes sesiones, de modo que en la base de datos las clases se almacenarán solo con su nombre y su número de filas y columnas.

En futuras versiones se podría incorporar la posibilidad de eliminar filas y columnas a una clase determinada, ya que así se evitaría la creación de una clase nueva cada vez que se haga una redistribución de los ordenadores en el aula.

² Si no existen tareas creadas, el usuario puede omitir este campo y la clase se lanzará sin contenido.

La pantalla de “selección de clase existente” es una versión simplificada de la de “creación de clase”, ya que dispone de un formulario con solo dos campos. En el primero el profesor selecciona una de las clases almacenadas en la base de datos, y, en el segundo, selecciona una tarea para asignarle a la sesión.

4.3.7. Mapa de la clase

Una vez abierta la clase, aparece la pantalla con el mapa de los puestos de los alumnos. En primer lugar aparecerán todos los sitios vacíos y más adelante se irán ocupando según seleccione su puesto cada alumno. Con este mapa se muestra toda la información que requiere el profesor para realizar la visualización del progreso a tiempo real de la clase. En la figura 4.9 se observa un ejemplo con un mapa de una sesión con ocho puestos ocupados.

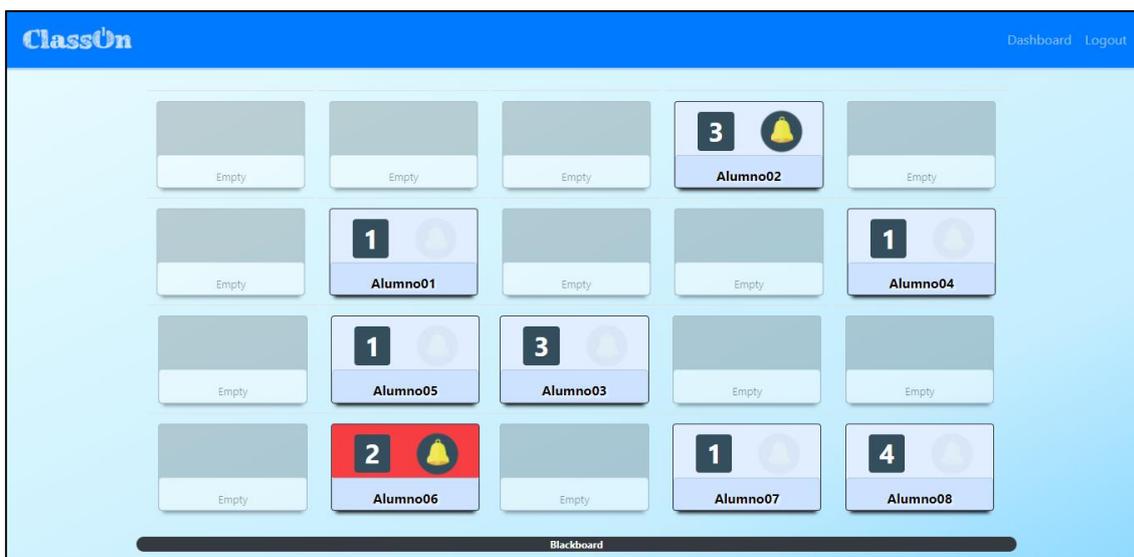


Figura 4.9. Mapa de una clase de 4x5 con ocho alumnos en ella

En la figura 4.9 se muestra una clase en la que ocho alumnos han seleccionado un puesto (en este ejemplo solo hay un alumno por ordenador). Los puestos sin ocupar permanecen en un color gris para no llamar la atención del profesor pero permitiendo a éste ubicarse correctamente en la clase. El aula del ejemplo dispone de cuatro filas y cinco columnas, por lo que se generan un total de veinte tarjetas. Debajo del mapa se coloca un

componente que representa la ubicación de la pizarra, de esta forma se crea un punto de referencia fijo en cada aula.

En cada tarjeta se muestra el nombre del alumno, el número del capítulo por el que va y una campana que indica si el alumno tiene una duda. En el caso de ejemplo, se observa que tanto *Alumno02* como *Alumno06* tienen una duda. La tarjeta de *Alumno06* aparece en rojo porque es la duda que le corresponde atender al profesor.

Es fundamental que no se modifique la ubicación de cada puesto en función de las dimensiones de la pantalla del profesor, ya que la estructura es clave para que el docente tenga un mapa correcto de los puestos. Además, el objetivo es que todos los puestos permanezcan visibles en la pantalla sin necesidad de hacer scroll, con el fin de garantizar la efectividad del seguimiento visual. Para conseguir esto, las tarjetas deben variar su tamaño en función de dos aspectos; las dimensiones de la pantalla y el número de filas y de columnas.

Contenido de las tarjetas

Debido a que las tarjetas modifican sus dimensiones, su contenido se verá afectado. Cuando las tarjetas reducen su tamaño, los componentes de su interior deberán adaptarse, siendo a veces necesario sacrificar parte de la información si existe un gran número de filas o columnas. Por el contrario, si las tarjetas aumentan de tamaño, los componentes de su interior pueden hacerse más grandes para facilitar el seguimiento visual por parte del profesor.



Figura 4.10. Tarjetas en función de su ancho

En la figura 4.10 se muestran los distintos tipos de tarjetas que existen en función del ancho. Cuando el ancho de la tarjeta es inferior a 100 píxeles, se sacrifica el icono de la campana, ya que para mantener ambos iconos hay que reducir en exceso su tamaño y estropean el seguimiento visual del profesor. Pese a no aparecer la campana, la tarjeta se seguirá poniendo con un fondo rojo para el alumno que debe ser atendido el siguiente por el profesor. El nombre del alumno también se irá cortando en función del ancho de la tarjeta, colocando una elipsis para mostrar que el nombre está incompleto.



Figura 4.11. Tarjetas en función de su altura

En la figura 4.11 se muestran dos tipos de tarjetas en función de la altura. La primera no tiene ningún tipo de restricción y muestra la altura máxima que puede tomar una tarjeta. La segunda muestra el tamaño mínimo para la altura de una tarjeta, ya que si se reduce más los iconos se hacen demasiado pequeños para el seguimiento visual. Se suprime primero el nombre del alumno porque es más prescindible que los otros dos elementos.

Tras hacer el desarrollo, se establece el número máximo de filas y columnas a ocho, puesto que un número mayor supone una reducción excesiva del tamaño de las tarjetas y el seguimiento visual resulta inviable en dispositivos con un ancho de pantalla menor.

El caso límite se da por tanto en tablets (1024px de ancho) con aulas de 8x8 y todos los puestos ocupados. Esto no supone un problema ya que la mayoría de las aulas tienen dimensiones menores. En la figura 4.12 se muestra un ejemplo del resultado de este caso.

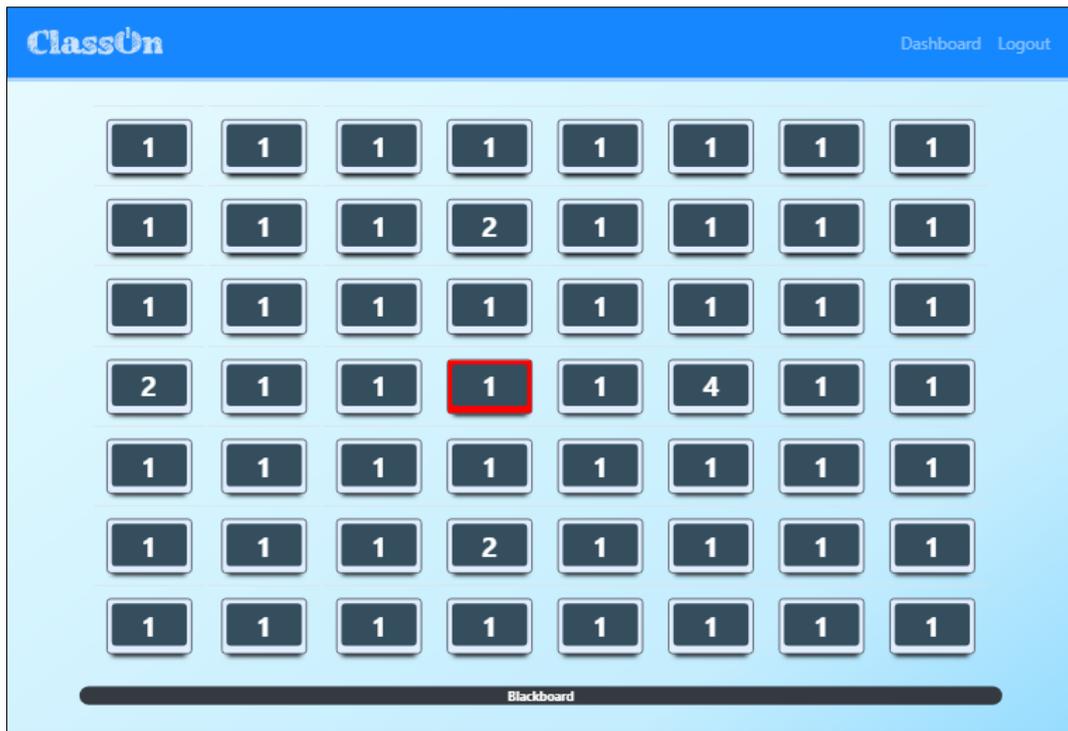


Figura 4.12. Ejemplo de mapa con un caso límite

Información detallada

Cuando el profesor pincha en una de las tarjetas le aparece un pop-up (cuadro de diálogo) con la información detallada de los alumnos de dicho puesto. En dicha información aparecerá la foto (si disponen de ella), su progreso, sus nombres completos y sus dudas pendientes de resolver. En la figura 4.13 se muestra un ejemplo con la información detallada de dos alumnos.

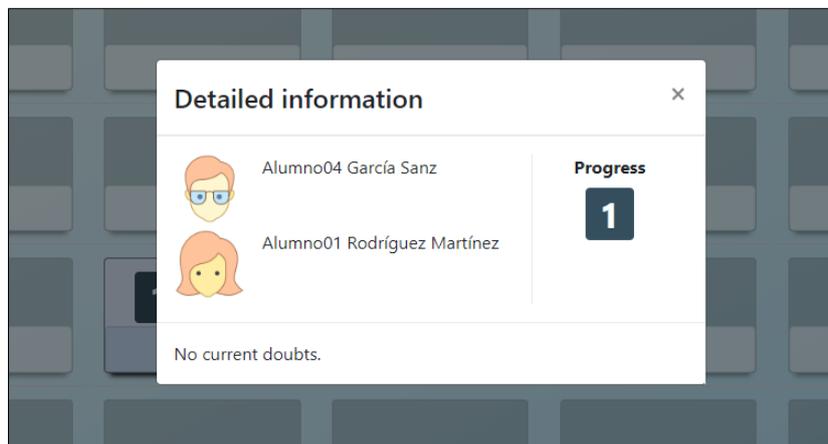


Figura 4.13. Información detallada de un puesto con dos alumnos

Cuando salta el pop-up con la información detallada, el resto de la pantalla se difumina, facilitando así la lectura de los datos. En este caso, las imágenes utilizadas son iconos de una librería asignados aleatoriamente debido a que ninguno de los estudiantes tiene una imagen registrada en ClassOn.

Para cerrar la información detallada basta con pinchar fuera del cuadro, de esta forma se agiliza la navegación del profesor que no tiene necesariamente que pinchar en la X de la esquina para cerrarlo.

Tablón de dudas

Debajo del mapa de la clase, aparecen las dudas que han subido los alumnos a la aplicación. Aquí, las dudas aparecen por orden de creación, por lo que no depende de la sección a la que correspondan o del número de dudas del alumno.

Dentro de cada duda, aparece información sobre la sección a la que corresponde, el alumno que la ha subido y el contenido de la propia duda. Además, aparece para cada duda un botón para marcarla como resuelta. Mediante este botón, el profesor elimina dicha duda cuando considera que ha sido resuelta. En la figura 4.14 se observa un ejemplo en el que dos alumnos han subido tres dudas respecto a dos secciones diferentes.



Figura 4.14 Tablón de dudas debajo del mapa de la clase

El profesor puede tomar aquí la decisión de contestar las preguntas en el orden en el que han sido mandadas, o seguir la recomendación de la aplicación (duda en rojo), que tiene en cuenta si el alumno ya ha recibido atención durante la sesión y el número de la sección de la duda.

El sistema de recomendación pretende dar prioridad a los alumnos que han sido atendidos menos veces de forma que haya un reparto justo de la atención del profesor, por lo que este es el criterio con más relevancia en la recomendación.

Si dos alumnos han sido atendidos el mismo número de veces, y hay una diferencia de dos o más secciones entre ellos, entonces se dará prioridad al alumno que vaya más atrás en el progreso de la tarea. Si la diferencia entre secciones es menor que dos, entonces se da preferencia al alumno que subió su duda primero.

4.3.8. Creación de tareas

La tercera opción en el panel de herramientas del profesor es la de creación de tareas. Mediante esta herramienta el profesor añadirá el contenido que les aparecerá a los alumnos durante cada clase, organizando el temario en secciones.

Lo primero que le aparece al profesor es un formulario con dos campos, uno para el nombre de la asignatura y otro para el nombre de esa tarea. Tras rellenar correctamente estos campos accederá a una nueva pantalla, mostrada en la figura 4.15, donde se podrán ir añadiendo las secciones a dicha tarea.

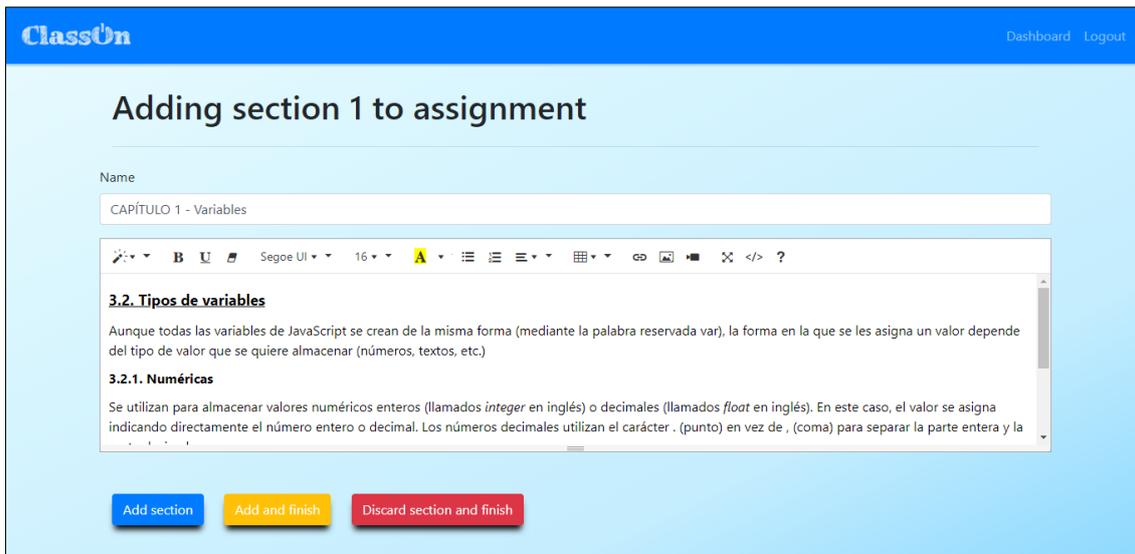


Figura 4.15. Pantalla para añadir secciones a una tarea

En esta pantalla se utiliza un tipo de formulario desarrollado por Summernote. Este formulario permite al usuario añadir formato al texto o añadir directamente código mediante un editor HTML.

Mediante los botones, el usuario puede añadir la sección y pasar a una nueva, añadir la sección y terminar la creación de la tarea o descartar la sección actual y terminar la creación de la tarea. Cuando se añade una sección, se almacena el código HTML en la base de datos MySQL, de forma que luego al estudiante le aparece la tarea con el formato con el que se ha diseñado.

Existe una limitación sin embargo a la hora de añadir imágenes o vídeos a la tarea ya que no se ha conseguido almacenar los archivos mediante este formulario.

4.3.9. Panel del estudiante

Tras terminar la explicación de las pantallas a las que accede el profesor, ahora se procede a explicar las pantallas a las que accede el usuario cuando es un alumno.

Al igual que el profesor, el alumno accede a un panel o *dashboard*, desde el que accederá a las demás pantallas. En el caso del alumno, su panel es más limitado, puesto que solo incluye una lista de las clases activas para seleccionar la que le corresponde en su clase. En la figura 4.16 se muestra el panel de un alumno con tres clases disponibles.

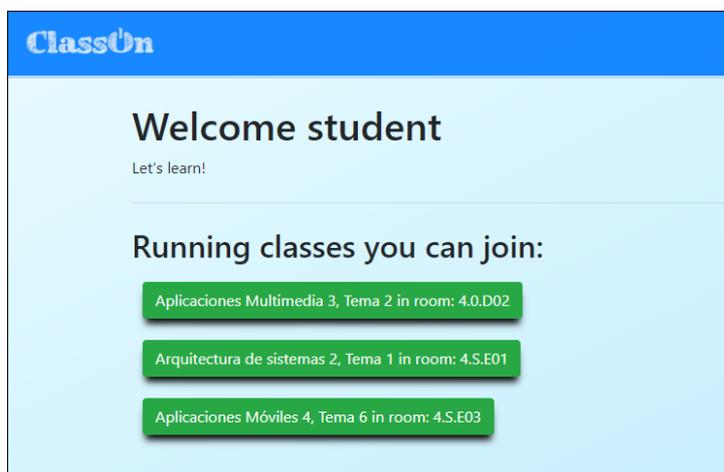


Figura 4.16. Recorte de la pantalla con el panel del estudiante

Tras seleccionar su clase, el estudiante accederá a un mapa con los distintos puestos del aula con la misma estructura que el mapa del profesor. En esa pantalla, mostrada en la figura 4.17 el alumno deberá ubicar su puesto y seleccionarlo en el mapa.

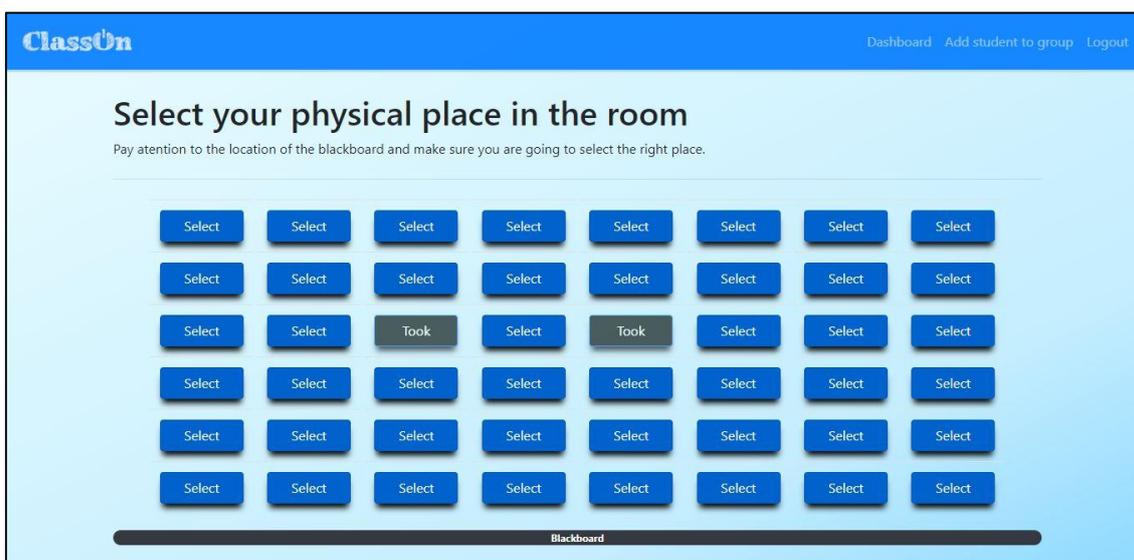


Figura 4.17. Pantalla en la que el estudiante selecciona su puesto

Otra opción que se barajaba inicialmente era que se detectara automáticamente el aula cuando un alumno inicia sesión en un puesto. Esta opción se descartó porque esta funcionalidad dejaría fuera a los alumnos que trabajan con su propio portátil o tablet.

4.3.10. Vista de tareas del estudiante

Tras seleccionar su puesto en el aula, el estudiante accede a la tarea asignada por el profesor para esa sesión. En la figura 4.18 se muestra la pantalla con el temario al que accede el alumno.

En este ejemplo, se utiliza la tarea creada en el capítulo 4.3.8 y se observa que se mantiene el formato que se estableció. En este caso se trata de una tarea con solamente dos secciones. El alumno navegará por ambas según lo necesite mediante los botones, siendo el número de la sección el que marcará su progreso en la tarea.

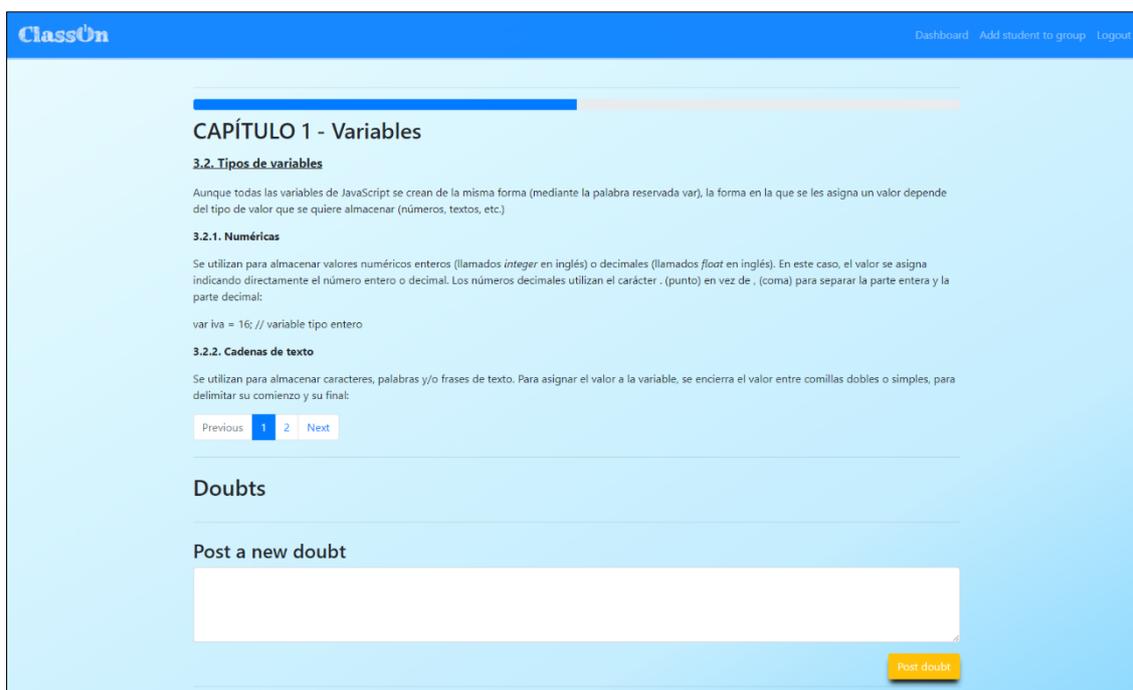


Figura 4.18. Pantalla de la vista de tareas para el estudiante

Mediante el formulario de abajo, el alumno publica las dudas que tenga respecto a esa sección. Es importante que suba la duda en la sección correspondiente, ya que la duda queda registrada en esa sección para ayudar a la gestión del profesor.

En la figura 4.18 también se observa como aparece un nuevo componente en la *Action bar* que permite al usuario agregar otro alumno a ese puesto. Al pinchar ahí, el estudiante es redirigido a una pantalla con un formulario para iniciar sesión, por lo que el nuevo estudiante debe estar ya registrado previamente.

Tablón de dudas

En cada sección el alumno puede ver las dudas creadas por sus compañeros asociadas a dicha sección. Aparecerán en el mismo orden en el que fueron creadas. En la figura 4.19 se muestra un ejemplo de una sección con dos dudas publicadas.

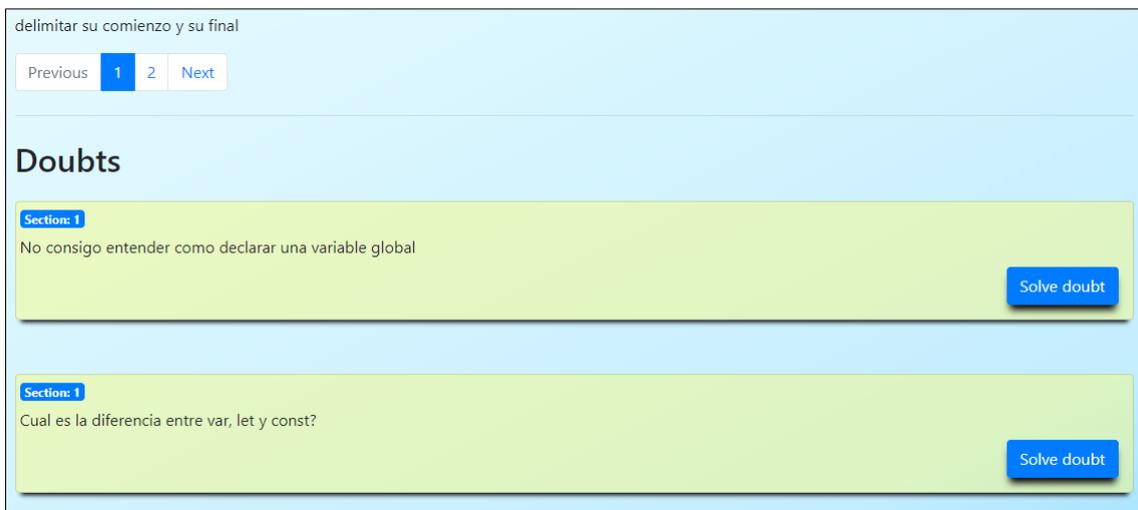


Figura 4.19. Tablón de dudas en la vista de tareas del alumno

En este tablón, se omite el nombre del alumno que ha subido la duda, para que los estudiantes no respondan en función de la persona que ha preguntado y así todos se ayuden entre sí.

Al pinchar en resolver duda, aparecerá un pop up en el que pueden escribir una respuesta para la pregunta de su compañero. Esta funcionalidad sin embargo no se ha conseguido desarrollar por problemas con el funcionamiento del back-end. La respuesta por tanto no llega al alumno que la planteó, lo cual supone una limitación que habría que resolver en versiones futuras. En la figura 4.20 se muestra un ejemplo del pop-up para resolver dudas.

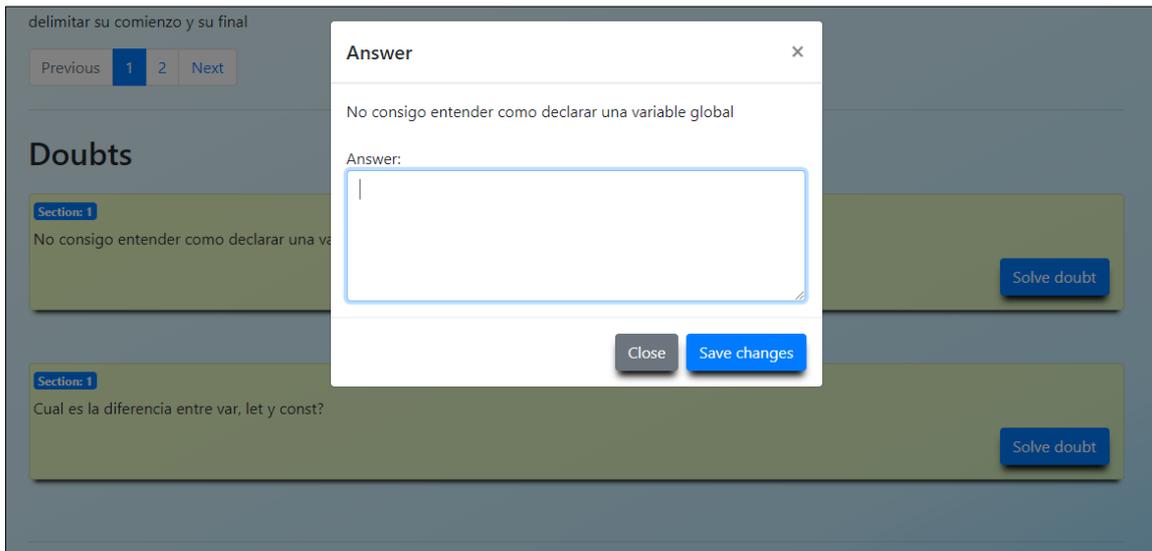


Figura 4.20. Pop-up para que el alumno responda dudas de sus compañeros

5. GESTIÓN DEL PROYECTO

5.1. Planificación del proyecto

En la tabla 5.1 se muestran las distintas fases con sus respectivas tareas (algunas realizadas a la vez que otras) y su duración en días.

TAREA	Fecha inicio	Nº Días	Fecha final
Introducción al proyecto y a su estructura	21-6-2018	30	21-7-2018
Análisis del proyecto y sus requerimientos	21-6-2018	2	23-6-2018
Conocer la estructura del back-end y su funciones	24-6-2018	27	21-7-2018
Leer documentación sobre versiones anteriores	24-6-2018	5	29-6-2018
Análisis del estado del arte de los métodos de aprendizaje	30-6-2018	13	13-7-2018
Análisis del estado del arte de las tecnologías utilizadas	14-7-2018	7	21-7-2018
Instalación de componentes necesarios y puesta a punto	22-7-2018	9	31-7-2018
Descarga de Pycharm (entorno Python) y librerías necesarias	22-7-2018	3	25-7-2018
Descarga de MySQL e inicialización	26-7-2018	1	27-7-2018
Ejecución del código y entender el funcionamiento del prototipo inicial	27-7-2018	4	31-7-2018
Desarrollo de la aplicación	15-8-2018	103	8-2-2019
Organización de la estructura y creación de repositorio	15-8-2018	4	19-8-2018
Pruebas en el entorno y bug fixing genérico	20-8-2018	10	30-8-2018
Desarrollo pantalla inicial	31-8-2018	4	4-9-2018
Mejora de los formularios de registro e inicio de sesión	5-9-2018	4	9-9-2018
Desarrollo panel del profesor	10-9-2018	4	14-9-2018
Mejora pantalla de creación de clase	15-9-2018	6	21-9-2018
Desarrollo del mapa de la clase	22-9-2018	15	7-10-2018
Desarrollo creación de tareas / vista de tareas del alumno	8-10-2018	12	20-10-2018
Desarrollo gestión de dudas	21-10-2018	14	4-11-2018
Bug fixing general y control de calidad	5-11-2018	14	19-11-2018
Instalación del servidor en una máquina virtual	1-2-2019	7	8-2-2019
Documentación del trabajo	1-11-2018	51	22-12-2018
Segunda fase de la documentación y revisión del trabajo	22-1-2019	24	15-2-2019

Tabla 5.1. Planificación del proyecto

Se ha dividido el proceso del trabajo en cinco fases. La primera comprende el tiempo dedicado a la familiarización con el proyecto y con las metodologías que aplica ClassOn. La segunda es la más corta porque únicamente incluye la instalación de los componentes

necesarios para el desarrollo de la aplicación. La tercera es con diferencia la más larga puesto que incluye todo el desarrollo del front-end y las nuevas funcionalidades del back-end.

La cuarta y la quinta fase han sido separadas porque hubo un intervalo de tiempo durante el cual, por motivos académicos, se detuvo el trabajo en este proyecto. Estas dos últimas fases incluyen la redacción de este informe y la revisión final.

En la fase de desarrollo, cada tarea incluye; análisis del requerimiento, desarrollo y comprobación del resultado. Se realiza una revisión general al finalizar la fase del desarrollo para comprobar que la aplicación funciona correctamente.

A lo largo de todo el proyecto, se ha contado con el trabajo de una directora de proyecto, que ha proporcionado su ayuda durante todas las fases. Además, para la fase de instalación del servidor, ha colaborado un técnico de sistemas de la Universidad Carlos III de Madrid.

5.2. Tecnología y recursos requeridos

Para realizar este proyecto se han utilizado los siguientes recursos y equipos:

- **Hardware:**
 - Ordenador portátil ASUS K555 con Intel Core i7 y Windows 10. Usado en el desarrollo y en la redacción del informe.
 - Tablet Samsung Galaxy 2 para pruebas de la aplicación.
 - Servidor Linux de la universidad UC3M.

- **Software:**
 - Pycharm Community Edition 2018 1.4. IDE para desarrollo y compilación de Python.
 - Librerías Python requeridas por la aplicación (Flask, dataStructures, WTForms, passLib).

- MySQL 5.7. WorkBench. Instalado para el uso de bases de datos en la aplicación. También requería la instalación de Visual Studio.
- WinLess. Compilador de lenguaje Less a CSS.
- Sublime Text 3. Un editor de texto enfocado a código.
- Microsoft Office. Utilizado para la redacción del informe y la creación de tablas y contenidos.

5.3. Presupuesto

Primero se evaluarán los costes directos generados durante la creación de la aplicación, tanto por personal como por el coste de depreciación del equipo utilizado.

Para la creación de esta herramienta se ha requerido el trabajo de tres ingenieros de la Universidad Carlos III de Madrid. En la Tabla 5.2 se detalla el presupuesto debido al coste del personal, siendo el coste total 7.920 €.

Nombre	Categoría	Salario por hora	Horas totales	Coste Total
Fernández, Sergio	Analista y desarrollador	17,50 €	320	5.600 €
Crespo, Raquel	Directora de proyecto	35 €	64	2.240 €
Rubio, Eugenio	Técnico de sistemas	20 €	4	80 €

Tabla 5.2. Costes debidos al personal

Es necesario también tener en cuenta el coste debido a la depreciación del equipo utilizado, es decir, la disminución de su precio debida a su uso en este proyecto. Para calcularlo se utiliza la siguiente ecuación que mide la depreciación de un equipo: [41]

$$\text{Coste Depreciación} = \frac{\text{Coste inicial} - \text{Coste residual}}{\text{Vida útil (horas)}} \cdot \text{Horas de uso}$$

En la Tabla 5.3. se detalla el coste debido a la depreciación de los equipos usados en ClassOn, resultando un total de 61 €.

Equipo	Coste inicial	Coste residual	Vida útil	Horas de uso	Coste total
Ordenador Asus	550 €	50,00 €	2500	300	60,00 €
Tablet Samsung	200 €	20 €	2000	10	0,90 €

Tabla 5.3. Costes debidos a la depreciación de los equipos.

Los costes generados equivalen ascenden por tanto a un total de 7.981 €.

5.4. Modelo de negocio

Al tratarse de una aplicación Open Source no se busca obtener ingresos mediante la venta directa del software o la incorporación de cuentas premium en la aplicación.

En la primera fase de ClassOn, esta será usada principalmente en la Universidad Carlos III de Madrid, donde se seguirá desarrollando y mejorando ante posibles errores de desarrollo. Una vez la aplicación esté lista y se compruebe su utilidad en situaciones reales, se lanzaría una segunda fase en la que la aplicación se subiría a una plataforma Open Source.

En esta segunda fase, se ofrecería la aplicación con todas las funcionalidades ilimitadas. En la aplicación se ofertaría un servicio extra de soporte y análisis de datos, el cual permitiría empezar a obtener ingresos. Si esta fase funciona correctamente, se contemplaría el pasar a una tercera fase en la que se incluiría una versión premium, limitando las funcionalidades gratuitas para empezar a obtener más ingresos.

5.5. Entorno socio-económico

Respecto al impacto socio-económico, el objetivo es que la aplicación ayude a profesores que puedan pertenecer a distintos ámbitos de enseñanza. Se trata de una aplicación gratuita que no requiere coste de ningún tipo para los centros, por lo que el precio no supondrá una barrera para su uso.

Está diseñada para mejorar el funcionamiento de las clases, sobre todo en las materias basadas en PBL, que se verán especialmente optimizadas con su uso. Se espera que el impacto social sea alto, y que los alumnos mejoren su experiencia en el aprendizaje, enriqueciéndose con la materia y desarrollando sus aptitudes de colaboración en equipo y de trabajo con herramientas digitales.

6. MARCO REGULADOR

En este capítulo se analiza la legislación aplicable sobre la aplicación ClassOn así como los estándares técnicos aplicados.

Al tratarse ClassOn de una aplicación open-source, se exige a ésta de obligaciones legales respecto a la comercialización del producto. Para que la aplicación sea open-source, se debe añadir una licencia de código abierto al repositorio de GitHub. Esta licencia es necesaria para que otras personas puedan usar, copiar, modificar y contribuir al proyecto de forma legal [42].

Para ClassOn se selecciona Apache 2.0 como licencia de código abierto. La licencia Apache [43] es una licencia de software libre permisiva que otorga al usuario del software libertad en su uso para cualquier propósito: distribución, modificación o comercialización. Esta licencia se incluye en los términos legales de la aplicación y puede comprobarse en el Anexo A de esta memoria.

En cuanto a privacidad, se aplica el nuevo Reglamento europeo de Protección de Datos (RGPD), un reglamento que protege los datos del usuario y regula su circulación [44]. Para cumplir con este reglamento, se tienen en cuenta los siguientes aspectos [45]:

- Se solicitan únicamente los datos personales necesarios para la aplicación. En este caso, nombre y apellidos, dirección de correo y fotografía de perfil.
- Se solicita consentimiento al usuario para tratar sus datos personales mediante la marcación de una casilla con la información necesaria.
- Dichos datos solo se utilizan dentro de la aplicación para fines educativos. Se podrán utilizar también con fines de investigación, siendo los datos previamente anonimizados.
- La aplicación dispondrá de información sobre el motivo del uso de dichos datos, los plazos de su conservación, las personas que tienen acceso a ellos y los derechos del usuario ante la gestión de sus datos.

- En esta primera versión de la aplicación solo se permitirá el uso de la aplicación para mayores de edad para evitar posibles problemas con el uso de datos de menores.
- Se incluirá en la pantalla *Home* el aviso legal y la política de privacidad con la información necesaria requerida en el reglamento.

Estándares técnicos

En desarrollo web, las especificaciones técnicas para HTML5 y CSS3 son gestionadas por el Consorcio World Wide Web (W3C) [28]. Un consorcio que se encarga de crear un consenso en la tecnología web utilizada y en garantizar su calidad.

También se utiliza el estándar ECMAScript 2016 [46], utilizado como estándar para JavaScript. ECMAScript es el estándar más utilizado en el lado del cliente en desarrollo web y cada vez es más usado en el lado del servidor.

7. CONCLUSIONES Y TRABAJO FUTURO

7.1. Conclusiones del proyecto

En este Trabajo de Fin de Grado se ha conseguido finalizar el desarrollo de una nueva versión de ClassOn que cumple con los objetivos planteados inicialmente. La aplicación resuelve los problemas mencionados en la introducción, otorgando al docente la capacidad de conocer el progreso de cada alumno en tiempo real y mejorando la gestión de las dudas de los alumnos. El único punto negativo ha sido que no se ha conseguido corregir el error que no permite que las respuestas de los alumnos lleguen a sus compañeros.

Esta versión de ClassOn cumple con los requerimientos de ser una aplicación intuitiva, sencilla y efectiva técnicamente. Además, la aplicación funciona correctamente en tablets y en ordenadores con distinto tamaño de pantalla, luego se consigue que sea adaptativa.

Se ha cumplido también el sub-objetivo de realizar un análisis de las metodologías de enseñanza en las que se basa ClassOn y de las soluciones tecnológicas que aplica.

Además, pese a que este trabajo no ha realizado el desarrollo completo del back-end, ha sido necesario entender como está implementado y conocer la estructura del framework Flask, ya que se han realizado modificaciones y se han añadido nuevas funcionalidades.

Para la redacción de este informe también ha sido necesario investigar sobre el marco regulador que se aplica en esta aplicación así como realizar un análisis del entorno socio-económico del proyecto.

7.2. Trabajo futuro

ClassOn es una aplicación a la que se le pueden incorporar muchas funcionalidades nuevas para seguir mejorando su funcionamiento. Con la versión

elaborada en este proyecto, se incorporan características nuevas y se consigue una aplicación funcional que se pretende utilizar en clases reales. El hecho de que se pueda empezar a usar es fundamental para su continuo desarrollo, puesto que en un caso real puede que aparezcan errores no localizados durante el desarrollo y tanto profesorado como alumnado pueden comentar sus experiencias en el uso de la aplicación.

A continuación se comentarán algunas funcionalidades que podrían incorporarse en futuras versiones de ClassOn:

- Incorporar la traducción a otros idiomas mediante la herramienta Poedit.
- Posibilidad de que los profesores se registren con un código de seguridad para que los alumnos no puedan crearse una cuenta de profesor.
- Opción para recuperar la contraseña en caso de que el usuario la olvide.
- Asignación automática de los sitios en la clase a partir de la IP de su puesto.
- Posibilidad de añadir un código de seguridad para acceder a cada clase para evitar que entren alumnos que no sean del grupo.
- Permitir la opción de dibujar el mapa de la clase una vez creada, de forma que el profesor pueda eliminar puestos del mapa e incluso modificar la ubicación de cada puesto mediante tecnología *drag and drop*.
- Almacenamiento del tiempo que permanecen los estudiantes en cada pantalla y el tiempo que esperan a ser atendidos. Con estos datos se podría hacer un análisis posterior muy útil para el docente mediante *learning analytics*.
- Posibilidad de añadir imágenes en las tareas y que estas se almacenen en el servidor.
- Resolver el error que no permite que las respuestas de los alumnos lleguen a sus compañeros.
- Mejorar el algoritmo de recomendación de atención de las dudas. Se podría incluso incorporar tecnología de *machine learning* para optimizar su funcionamiento.

BIBLIOGRAFÍA

- [1] R. T. Johnson and D. W. Johnson, “Active Learning: Cooperation in the Classroom,” 2008.
- [2] D. L. Ball and A. F. Thurnau, “Strategies for Engaging Students in Large Classes.”
- [3] P. C. Blumenfeld, E. Soloway, R. W. Marx, J. S. Krajcik, M. Guzdial, and A. Palincsar, “Motivating Project-Based Learning: Sustaining the Doing, Supporting the Learning,” *Educ. Psychol.*, vol. 26, no. 3–4, pp. 369–398, Jun. 1991.
- [4] E. Tambouris *et al.*, “Enabling Problem Based Learning through Web 2.0 Technologies: PBL 2.0,” *Educ. Technol. Soc.*, vol. 15, pp. 238–251, 2012.
- [5] C. Campbell, “Problem-based learning and project-based learning,” 16-Sep-2014. [Online]. Available: <https://www.teachermagazine.com.au/articles/problem-based-learning-and-project-based-learning>. [Accessed: 20-Nov-2019].
- [6] G. M. Novak, “Just-in-time teaching,” *New Dir. Teach. Learn.*, vol. 2011, no. 128, pp. 63–73, Dec. 2011.
- [7] A. P. Fagen, C. H. Crouch, and E. Mazur, “Peer Instruction: Results from a Range of Classrooms,” *Phys. Teach.*, vol. 40, no. 4, pp. 206–209, Apr. 2002.
- [8] “Web 2.0 Technologies and Back Channel Communication in an Online Learning Community,” *TechTrends*, vol. 54, no. 4, pp. 41–51, Jul. 2010.
- [9] S. Yardi, “Digital Youth, Innovation, and the Unexpected,” pp. 143–164, 2008.
- [10] A. Pohl, F. Bry, and V. Gehlen-Baum, “Introducing Backstage – a digital backchannel for large class lectures,” *Interact. Technol. Smart Educ.*, 2011.
- [11] A. Pohl, F. Bry, J. Schwarz, and M. Gottstein, “Sensing the classroom: Improving awareness and self-awareness of students in Backstage,” in *2012 15th International Conference on Interactive Collaborative Learning (ICL)*, 2012, pp. 1–8.
- [12] “Google Classroom.” [Online]. Available: <https://classroom.google.com/>. [Accessed: 05-Dec-2018].
- [13] “Spark learning with G Suite for Education.” [Online]. Available: https://edu.google.com/products/gsuite-for-education/?modal_active=none.

[Accessed: 01-Dec-2018].

- [14] “Welcome to your first day of Classroom.” [Online]. Available: <https://teachercenter.withgoogle.com/first-day-trainings/welcome-to-classroom>. [Accessed: 01-Dec-2018].
- [15] I. N. M. Shaharane, J. M. Jamil, and S. S. M. Rodzi, “Google classroom as a tool for active learning,” *AIP Conf. Proc.*, vol. 1761, 2016.
- [16] R. Joy *et al.*, “Usability Evaluation of Google Classroom: Basis for the Adaptation of GSuite E-Learning Platform,” *Asia Pacific J. Educ. Arts Sci.*, vol. 5, no. 1, pp. 47–51, 2018.
- [17] L. Dvorkin Camiel, J. D. Goldman-Levine, M. D. Kostka-Rokosz, and W. W. McCloskey, “Twitter as an In-Class Backchannel Tool in a Large Required Pharmacy Course,” *Am. J. Pharm. Educ.*, vol. 78, 2014.
- [18] R. Kimmons and G. Veletsianos, “Education scholars’ evolving uses of twitter as a conference backchannel and social commentary platform,” in *British Journal of Educational Technology*, 2016.
- [19] V. J. Shute, “Focus on Formative Feedback,” *Rev. Educ. Res.*, vol. 78, no. 1, pp. 153–189, 2008.
- [20] H. S. Alavi and P. Dillenbourg, “An ambient awareness tool for supporting supervised collaborative problem solving,” *IEEE Trans. Learn. Technol.*, vol. 5, no. 3, pp. 264–274, 2012.
- [21] R. Martinez-Maldonado, A. Clayphan, K. Yacef, and J. Kay, “MTFeedback: Providing notifications to enhance teacher awareness of small group work in the classroom,” *IEEE Trans. Learn. Technol.*, vol. 8, no. 2, pp. 187–200, 2015.
- [22] K. Holstein, G. Hong, M. Tegene, B. M. McLaren, and V. Alevan, “The classroom as a dashboard: Co-designing Wearable Cognitive Augmentation for K-12 Teachers,” *Proc. 8th Int. Conf. Learn. Anal. Knowl. - LAK '18*, no. c, pp. 79–88, 2018.
- [23] T. Zarraonandia, I. Aedo, P. Díaz, and A. Montero, “An augmented lecture feedback system to support learner and teacher communication,” *Br. J. Educ. Technol.*, vol. 44, no. 4, pp. 616–628, 2013.

- [24] “Socrative.” [Online]. Available: <https://www.socrative.com/>. [Accessed: 01-Dec-2018].
- [25] I. Gutiérrez Rojas, R. M. Crespo García, and C. Delgado Kloos, “Orchestration and feedback in lab sessions: Improvements in quick feedback provision,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2011.
- [26] I. Gutiérrez Rojas, R. M. Crespo García, and C. Delgado Kloos, “Enhancing Orchestration of Lab Sessions by Means of Awareness Mechanisms,” 2012.
- [27] I. Gutiérrez-Rojas, R. M. Crespo-García, and C. D. Kloos, “Adapting an awareness tool for massive courses: The case of classON,” *J. Univers. Comput. Sci.*, 2014.
- [28] “Standards - W3C.” [Online]. Available: <https://www.w3.org/standards/>. [Accessed: 03-Feb-2019].
- [29] “Less.js.” [Online]. Available: <http://lesscss.org/>. [Accessed: 03-Feb-2019].
- [30] “Flask (A Python Microframework).” [Online]. Available: <http://flask.pocoo.org/>. [Accessed: 03-Feb-2019].
- [31] “MySQL.” [Online]. Available: <https://www.mysql.com/>. [Accessed: 03-Feb-2019].
- [32] “About - Git.” [Online]. Available: <https://git-scm.com/about>. [Accessed: 10-Feb-2019].
- [33] G. Siemens and R. S. J. d. Baker, “Learning analytics and educational data mining,” in *Proceedings of the 2nd International Conference on Learning Analytics and Knowledge - LAK '12*, 2012, p. 252.
- [34] “Flask-SocketIO.” [Online]. Available: <https://flask-socketio.readthedocs.io/en/latest/>. [Accessed: 02-Feb-2019].
- [35] “Flask-MySQL.” [Online]. Available: <https://flask-mysql.readthedocs.io/en/latest/>. [Accessed: 02-Feb-2019].
- [36] “Flask-WTF.” [Online]. Available: <https://flask-wtf.readthedocs.io/en/stable/>. [Accessed: 02-Feb-2019].
- [37] “WinLess - Windows GUI for less.js.” [Online]. Available: <http://winless.org/>. [Accessed: 30-Dec-2019].

- [38] “Bootstrap.” [Online]. Available: <https://getbootstrap.com/>. [Accessed: 01-Feb-2019].
- [39] “Summernote.” [Online]. Available: <https://summernote.org/>. [Accessed: 01-Feb-2019].
- [40] “Icons8.” [Online]. Available: <https://icons8.com/icons>. [Accessed: 01-Feb-2019].
- [41] M. Kaelin, “How to calculate depreciation on computer hardware: A cheat sheet,” 2017. [Online]. Available: <https://www.techrepublic.com/article/how-to-calculate-depreciation-on-computer-hardware-the-smart-persons-guide/>. [Accessed: 31-Jan-2019].
- [42] “Aspectos legales del código abierto. | Open Source Guides.” [Online]. Available: <https://opensource.guide/es/legal/>. [Accessed: 02-Feb-2019].
- [43] “The Apache HTTP Server Project.” [Online]. Available: <https://httpd.apache.org/>. [Accessed: 10-Feb-2019].
- [44] General Data Protection Regulation, “Reglamento (UE) 2016/679 del parlamento europeo y del consejo de 27 de abril de 2016 relativo a la protección de las personas físicas en lo que respecta al tratamiento de datos personales y a la libre circulación de estos datos y por el que se deroga la D,” *D. Of. la Unión Eur.*, vol. 2014, no. 119, pp. 1–88, 2016.
- [45] “RGPD ¿Cómo cumplir el Reglamento Europeo de Protección de Datos?” [Online]. Available: <https://ayudaleyprotecciondatos.es/guia-rgpd/>. [Accessed: 02-Feb-2019].
- [46] “ECMAScript 2016.” [Online]. Available: <https://www.ecma-international.org/ecma-262/7.0/>. [Accessed: 10-Feb-2019].

ANEXO A – TÉRMINOS LEGALES

Licensed under the Apache License, Version 2.0 (the "License"); you may not use this file except in compliance with the License. You may obtain a copy of the License at:

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

ANEXO B – ENGLISH SUMMARY

Technology is advancing daily and most of the fields get benefit with the new digital tools and facilities that technology offers. In the field of teaching, using these technology tools improves significantly the learning experience. For example, there are tools which may promote collaboration and interaction between students, optimize teacher's work time or just host academic material.

In this project, technology will be used in order to improve the learning experience. It is pretended to resolve two main problems that are usually generated in practical sessions; bad management of students' doubts and teacher's lost of awareness of students' progress. In order to resolve these problems, **ClassOn** was created. ClassOn is a web tool which includes these characteristics:

- It helps to improve teacher's management of the doubts with a system that allows students to upload doubts and send them directly to the tether. Also, it helps the teacher to select the correct order to answer those doubts.
- It provides the teacher awareness tools in order to give him real time information about students' progress.

The main objective of this project is developing ClassOn new version. The tool must include the two previously described characteristics and it must be an intuitive, effective and adaptive application.

ClassOn functioning is based on different learning methodologies. The main ones are active learning and collaborative learning.

- **Active learning:** it is a form of teaching that involves students in the learning process and make them participate in the explanation instead of making them remain passive during the whole lesson.

- **Collaborative learning:** it is based on organizing the students into small groups in order to make them achieve together learning goals.

There are teaching strategies that are based on these two methodologies; **project based learning** and **problem based learning** (PBL). With these strategies, small group of students must do a project or resolve several problems that are related to the subject in order to get knowledge and improve their cooperation and problem-resolving aptitudes. ClassOn is created as a tool that focus in these PBL strategies but it might also be used with other teaching methods.

ClassOn makes use of two technology solutions in order to achieve the objectives:

- **Backchannel:** technology solution which is based on the creation of a real time dialog between the teacher and the students in order to establish a parallel conversation related to the subject. This may be done with the use of a digital wall or a real time chat. The main objective of backchannel is improve interaction and promote collaborative learning.

There are some related tools that make use of backchannel: Backstage, Google Classroom and Twitter (with hashtags). The three of them come to a positive conclusion but it is also added that the use of backchannel may cause students to get distracted.

- **Information display:** information display is an awareness tool which help the teacher to know the progress of every student in the classroom. Using this awareness tool, the teacher gain knowledge about the state of the class which it is very important in order to improve the learning process.

Some tools that use similar awareness tools are Lantern and Lumilo which give real time information to the teacher about students' progress.

ClassOn must include backchannel and information display in order to mitigate the learning problems described in the beginning. ClassOn is designed to be used in practical sessions at the computer lab because every student, or group of students, must be able to use a computer or tablet.

This web application is implemented according to client-server architecture which may be divided in two blocks; front-end and back-end. **Back-end** describes the server side of the application and it is also in charge of the communication with front-end and the database. **Front-end** describes the client side of the application, so it is in charge of developing the interface and providing functionality to the different components. The complete technology architecture is shown in figure S.1.

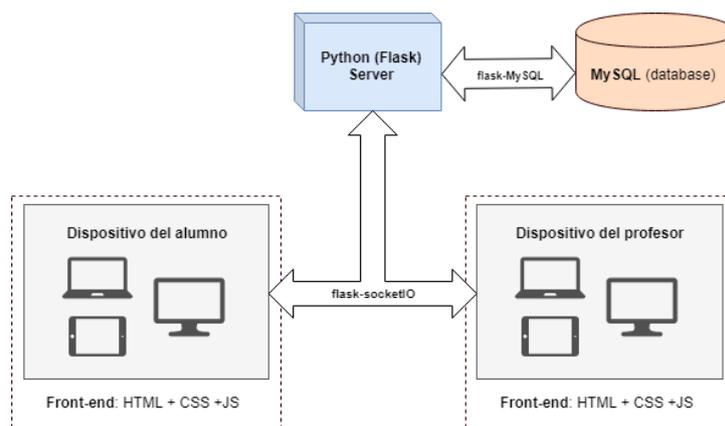


Figure S.1. ClassOn technology architecture

In this project, the new version of ClassOn will be developed starting at Jorge Frias' development of back-end. Some new functionalities are added in the back-end and also some bugs have been corrected. The front-end will be also developed starting at the initial prototype that J.Frias made.

The server side is developed with Python and Flask, a framework which allows web development with Python. Also, it includes some add-ons; flask-socketio, flask-mysqldb and flask-wtf. MySQL is the software used as the database. The client side is developed with HTML, CSS and JavaScript and it makes use of Less and Bootstrap as frameworks.

Now, I will describe the main screens and functionalities in the developed application:

- **Home:** this screen is the first one that the user watches. There are two main buttons, which allow the user to log in or to register. Both ‘log in’ and ‘register’ screens include a form that the user must complete. The user must select between two forms, one for students and one for teachers.
- **Teacher dashboard:** if user logs in correctly as a teacher, a new screen will appear, the teacher dashboard. In this screen there are three buttons, each one with one tool that the teacher may use. Those tools are ‘Create new classroom’, ‘Open existing classroom’ and ‘Create new assignment’.
- **Create classroom:** in this screen, teacher is allowed to create one classroom. To do this, he must complete a form with the name of the classroom and the number of columns and rows. The teacher also must select an assignment that will be opened with the classroom only the first time.
- **Classroom map:** once the classroom has been opened, a new screen will appear with that classroom map. First, all the places will be empty and then the students will appear in their places as soon as they select them.

The places will be represented with a card that contain the information of the student, including name, progress and a bell icon that alerts if the student has a doubt. An example of this map is shown at figure S.2

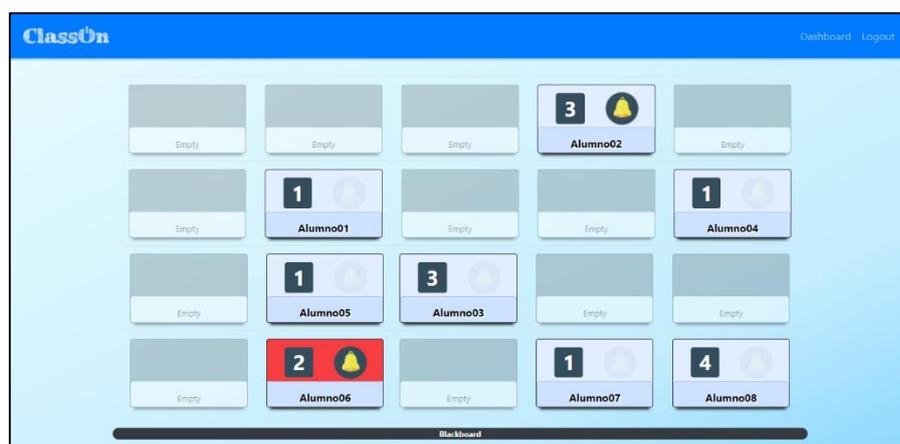


Figure S.2. Classroom map with eight students

When teacher clicks one card, a pop-up will appear with the student’s detailed information, including the content of his or her doubts.

Below this map, there is a doubts wall, in which all the doubts from the students appear in order.

- **Create assignment:** the third option in the teacher dashboard is “Create assignment”. This tool will allow the teacher to add content to a new assignment that will be shown to the students with the same format. The assignment is recommended to be separated into sections.
- **Student’s screens:** if the user logs in as a student, then he or she must select the classroom where they are. Then, a map will be shown so the user selects the place he or she is into.

After this, the student accesses to the assignment view, where it is shown the assignment that the teacher created previously. Below the material content, there is a form where the students are allowed to upload their doubts and also see other students’ doubts. An example is shown at figure S.3.

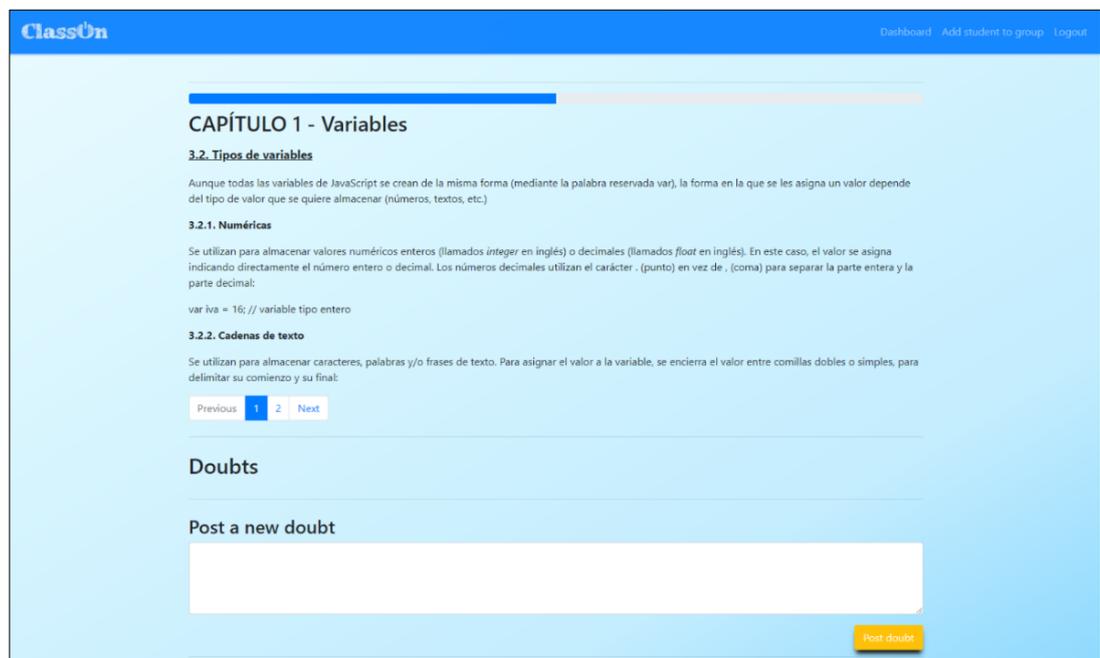


Figure S.3. Assignment screen for students

The process of this project has been divided into five stages:

1. Understanding the project and analyzing the requirements. Also, investigating about different learning methods and about technology at learning field.
2. Installation of the components that are required for the development.
3. Development of the application.
4. Writing this report.
5. Look over the whole project.

Regarding socio-economic impact, the main objective of this application is to help teachers and students from different learning fields. ClassOn is a free application so it does not require extra costs to the schools or universities.

This project causes economic costs due to the personal and equipment requirements. The total costs come to 7.981€.

ClassOn is an open-source application, so it avoids legal obligations about commercialization. Regarding privacy, the General Data Protection Regulation is applied for data use.

In conclusion, this project has achieved the objectives that were intended. The application resolves the problem that were described in the beginning, giving an awareness tool to the teacher and improving the management of the doubts.