



UNIVERSIDAD DE VALLADOLID

ESCUELA TÉCNICA SUPERIOR DE INGENIEROS DE
TELECOMUNICACIÓN

TRABAJO FINAL DE GRADO

GRADO EN INGENIERÍA EN TECNOLOGÍAS DE
TELECOMUNICACIÓN

APRENDIZAJE DE DISEÑO DE BASES DE DATOS
RELACIONALES A TRAVÉS DE MOODLE

MARZO, 2018

AUTOR: FERNANDO RAMÍREZ FELIÚ

TUTORA: MÍRIAM ANTÓN RODRÍGUEZ

TÍTULO: APRENDIZAJE DE DISEÑO DE BASES DE DATOS RELACIONALES A TRAVÉS DE MOODLE

AUTOR: FERNANDO RAMÍREZ FELIÚ

TUTORA: MÍRIAM ANTÓN RODRÍGUEZ

DEPARTAMENTO: TEORÍA DE LA SEÑAL, COMUNICACIONES E INGENIERÍA TELEMÁTICA

Miembros del Tribunal

PRESIDENTE: M^a ÁNGELES PÉREZ JUÁREZ

SECRETARIO: DAVID GONZÁLEZ ORTEGA

VOCAL: MÍRIAM ANTÓN RODRÍGUEZ

SUPLENTE: MARIO MARTÍNEZ ZARZUELA

SUPLENTE: FRANCISCO JAVIER DÍAZ PERNAS

FECHA DE LECTURA: Marzo de 2018

CALIFICACIÓN:

Resumen del proyecto

Este proyecto tiene como objetivo actualizar una herramienta muy necesaria para el alumnado y para los docentes. El trabajo principal ha sido la creación de un módulo para las nuevas versiones de Moodle, basándose en una herramienta anteriormente desarrollada y que ya no era funcional debido a su obsolescencia.

Dbquiz es una herramienta creada para el uso docente, tanto para el aprendizaje como en la evaluación de los alumnos. Se ha tratado de hacerla lo más interactiva y participativa, de modo que todos los usuarios se sientan cómodos desde el principio.

El fin principal del desarrollo de esta herramienta es dar soporte al equipo docente en las asignaturas Laboratorio de Desarrollo de Sistemas Telemáticos (LDST) y Tecnologías para Aplicaciones Web (TAW) y ayudar a los alumnos a mejorar sus conocimientos. Dbquiz construye ejercicios, ya sea para la práctica del alumnado o la evaluación de los conocimientos de los mismos, de construcción de base de datos relacionales y de la realización de consultas a estas. Las consultas se articulan sobre el lenguaje SQL.

A nivel de programación, se han utilizado los lenguajes empleados en la construcción de Moodle y, además, del uso de las librerías desarrolladas para la propia plataforma de Moodle.

Aunque no se trata de la ocupación inicial, se ha desarrollado una revisión de multitud de plataformas de e-learning, destacando sus ventajas e inconvenientes. Además, de un estudio de la plataforma Moodle empleada por la Universidad de Valladolid para construir su campus virtual.

Palabras clave

eLearning, Moodle, Dbquiz, bases de datos relacionales, SQL

Abstract

This project aims to update as much as needed tool for students and teachers. The main work has been the creation of a module for the new versions of Moodle, based on a previously developed tool that was no longer functional due to its obsolescence.

Dbquiz is a tool created for teaching use, both for learning and for the evaluation of students. It has tried to make it more interactive and participatory, so that all users feel comfortable from the start.

The main purpose of the development of this tool is to support the teaching team in the subjects Laboratorio de Desarrollo de Sistemas Telemáticos (LDST) y Tecnologías para Aplicaciones Web (TAW) and to help students to improve their knowledge. Dbquiz constructs exercises, either for the practice of the student or the evaluation of the knowledge of the same ones, of construction of relational databases and of the realization of consultations to these. The queries are articulated on the SQL language.

At the programming approach, the languages utilized in the construction of Moodle have been used, as well as the use of the libraries developed for the Moodle platform itself.

Although it is not about the initial occupation, a review of a multitude of e-learning platforms has been developed, highlighting its advantages and disadvantages. In addition, a study of the Moodle platform used by the University of Valladolid to build its virtual campus.

Key words

eLearning, Moodle, Dbquiz, Relational databases, SQL

AGRADECIMIENTOS

A mi familia, por su apoyo incondicional en los momentos difíciles.

A mi ella, por hacer más fácil la vida y centrarme en el camino.

A mis compañeros, por hacer de estos años hayan merecido la pena.

Y a mí tutora, por su ayuda en todo momento.

Gracias.

Índice

Resumen del proyecto	3
Abstract	4
Índice	6
Índice de figuras	8
Capítulo 1: Introducción	10
1.1. Introducción.....	10
1.2. Objetivos.....	10
1.3. Estructura.....	11
Capítulo 2: E-learning	12
2.1. ¿Qué es el e-learning?.....	12
2.2. Características del e-learning.....	12
2.3. Blended-Learning.....	13
2.4. ¿Qué es el LMS?.....	13
Capítulo 3: Moodle 3	28
3.1. Perfil de administrador.....	28
3.2. Perfil de profesor.....	32
3.3. Perfil de alumno.....	32
3.4. Consideraciones para desarrollar en Moodle.....	33
3.5. Tecnologías incluidas en Moodle.....	34
3.5.1. PHP.....	34
3.5.2. JavaScript.....	35
3.5.3. CSS.....	36
3.5.4. HTML.....	37
3.5.5. SQL.....	38
Capítulo 4: Módulo dbquiz	40
4.1. Introducción.....	40
4.2. Requisitos funcionales.....	40
4.2.1. Requisitos básicos.....	40
4.2.2. Requisitos técnicos.....	40
4.3. Diseño de la base de datos.....	41
4.4. Archivos del módulo.....	48
4.4.1. Archivos requeridos.....	48
4.4.2. Archivos de nueva creación.....	53
Capítulo 5: Manual de usuario	63
5.1. Instalación.....	63
5.2. Creación y configuración de una instancia.....	67
5.3. Creación de la base de datos.....	73
5.4. Edición de la base de datos.....	75

5.5. Creación de preguntas.....	75
5.6. Resolución del ejercicio.....	77
5.7. Correcciones.....	88
Capítulo 6: Estudio económico.....	92
Capítulo 7: Conclusiones.....	93
Capítulo 8: Líneas futuras.....	95
8.1. Edición de preguntas.....	95
8.2. Resolución del ejercicio.....	95
8.3. Calificaciones.....	96
8.4. Sección de estadísticas.....	96
8.5. Concepto de sandbox.....	96
Capítulo 9: Referencias.....	98

Índice de figuras

Ilustración 1: Portal de Almagesto	16
Ilustración 2: Portal de Blackboard	18
Ilustración 3: gestor de contenidos, Blackboard.....	19
Ilustración 4: Portal de e-ducativa	20
Ilustración 5: Portal de Chamilo.....	22
Ilustración 6: Portal de Moodle	25
Ilustración 7: Esquema de la base de datos de Dbquiz	42
Ilustración 8: Index del módulo dbquiz	49
Ilustración 9: Diagrama de flujo de view.php	50
Ilustración 10: Diagrama de flujo de edit.php	54
Ilustración 11: Diagrama de flujo de add_question.php	55
Ilustración 12: Diagrama de flujo del registro de tablas y campos	59
Ilustración 13: Diagrama de flujo de scoring.php	61
Ilustración 14: Directorio donde se aloja el módulo para ser instalado	63
Ilustración 15: instalación del módulo	64
Ilustración 16: Pestaña lateral de administración del sitio	64
Ilustración 17: Apartado para gestionar la instalación del módulo	65
Ilustración 18: Confirmación de instalación del módulo.....	65
Ilustración 19: Creación de una instancia.....	66
Ilustración 20: Apoyo gráfico de las recomendaciones en la instalación.....	67
Ilustración 21:Pestaña de activación de edición del curso.....	68
Ilustración 22: Curso preparado para crear una nueva instancia	69
Ilustración 23: Creación de la instancia.....	70
Ilustración 24: Cuadro de texto para incluir la descripción del ejercicio	71
Ilustración 25: Cuadro de texto para incluir las pistas en el ejercicio.....	71
Ilustración 26: Selectores para añadir fechas de apertura y cierre de ejercicio.....	71
Ilustración 27: Cuadros de texto donde añadir número máximo de tablas y campos	72
Ilustración 28: Cuadros de texto para añadir puntuación máxima y el número de intentos.....	72
Ilustración 29:Ajustes comunes de la instancia	72
Ilustración 30: Botones de guardado	72
Ilustración 31: Instancia creada en el curso	73
Ilustración 32: Redirección a edición de base de datos	73
Ilustración 33: Cuadro de texto para añadir el script de creación de base de datos.....	74
Ilustración 34: Script incluido en el cuadro de texto.....	74
Ilustración 35: Presentación de la base de datos añadida	75
Ilustración 36: Mensaje de edición de la base de datos	75
Ilustración 37: File manager	76
Ilustración 38: Cuadro de texto para añadir las cuestiones y su puntuación.....	76
Ilustración 39: Ejemplo de creación de preguntas.....	76
Ilustración 40: Visualización de las preguntas junto con la imagen.....	77
Ilustración 41: Visualización del curso con la instancia creada con el rol de alumno.....	77
Ilustración 42: Presentación del ejercicio al alumno	78
Ilustración 43: Cuadro de texto para añadir el número de tablas	78
Ilustración 44: Visualización del número máximo de tablas a incluir	79

Ilustración 45: Cuadros de texto para darle nombre a las tablas que se van a crear	79
Ilustración 46: Presentación de los campos y los selectores para incluir dichos campos en las tablas anteriormente creadas.....	80
Ilustración 47: Mensaje de error indicando que quedan campos por agregar a las tablas	80
Ilustración 48: Mensaje de error indicando que una tabla ha quedado vacía	81
Ilustración 49: Mensaje de error indicando que se ha superado el número máximo de campos que se pueden incluir en las tablas	81
Ilustración 50: Tablas creadas con sus campos añadidos	82
Ilustración 51: Cuadro de texto para añadir el número de relaciones entre tablas.....	82
Ilustración 52: Selección de las claves dentro de una tabla	83
Ilustración 53: Selectores para establecer relaciones entre tablas	83
Ilustración 54: Cuadros de texto para añadir las respuestas a las preguntas planteadas	84
Ilustración 55: Validación de la resolución del ejercicio	85
Ilustración 56: Botones de validación	86
Ilustración 57: Botones de validación	86
Ilustración 58: Presentación del ejercicio una vez realizado y a la espera de corrección	86
Ilustración 59: Presentación de la corrección de la base de datos	87
Ilustración 60: Presentación de la corrección de las respuestas	88
Ilustración 61: Visualización del curso con rol de administrador o profesor	88
Ilustración 62: Pestañas de navegación de la instancia	89
Ilustración 63: Listado de alumnos que han realizado el ejercicio y se les puede evaluar	89
Ilustración 64: Selección de alumno para evaluar	89
Ilustración 65: Presentación de la resolución del ejercicio con los cuadros de texto para añadir puntuación y comentarios.....	90
Ilustración 66: Presentación de las respuestas del alumno y los cuadros de texto para añadir puntuación y comentarios.....	91
Ilustración 67: Visualización de la puntuación que va a recibiendo cada alumno	91

1. Introducción

1.1. [Introducción](#)

En una sociedad tan cambiante como la actual donde estar a la vanguardia de las nuevas tecnologías es necesario tener estar en permanente contacto con el aprendizaje. A cualquier edad, todas las personas tienen la necesidad de aprender. Ya sea por requerimientos laborales, escolares o simplemente por curiosidad, la sociedad necesita tener acceso a información y material de aprendizaje.

Actualmente, debido a los tiempos de los que se dispone y de los avances de internet ha sido posible ir migrando toda esta infraestructura a la red. Propiciando así la máxima disponibilidad y abierto al gran público. Cada vez más aparecen nuevas plataformas donde realizar formación online. Hasta las propias universidades incentivan este uso haciendo disponibles sus cursos universitarios en plataformas de aprendizaje en red para que estudiantes de todo el mundo puedan acceder a ellos.

En 2017, se estima que más de la mitad de la población mundial dispone de acceso a internet. Esto supone que cerca de 4.000 millones de personas en todo el mundo podrán acceder a internet y hacer uso de estas plataformas. (UNESCO, 2014) Esto indica la importancia de disponer de contenido docente en la red.

1.2. [Objetivos](#)

Este proyecto radica en la necesidad de actualización de una herramienta ya usada anteriormente por el profesorado de las asignaturas de Laboratorio de Desarrollo de Sistemas Telemáticos (LDST) o Tecnologías para Aplicaciones Web (TAW). Esta herramienta apoya a los docentes a la hora de fijar los conocimientos impartidos de forma teórica.

Dicha herramienta debe poder estar disponible en modo online. Permitiendo ser usada por profesores y alumnos tengan acceso a ella en todo momento desde cualquier lugar. Debe ser una herramienta que fomente el aprendizaje, facilitando el uso para profesores y alumnos. Dotada de interactividad, lo que permite al alumno sentirse parte del aprendizaje. Esto supone que el alumno se involucre más en la actividad desarrollada durante el curso y le permita mejorar en sus resultados. Para el profesor debe suponer un avance en la forma de llegar a sus alumnos permitiéndole evaluar si fuese necesario desde la herramienta a los alumnos.

Por otro lado, se solicita hacer una revisión en profundidad de las distintas plataformas de aprendizaje en red disponibles actualmente. Se trata hacer una revisión de todo tipo de herramientas. Es necesario revisar desde las gratuitas a las plataformas de pago. Realizando un informe de puntos fuertes y puntos débiles para determinar si realmente la merece la pena utilizar alguna de ellas para desarrollar la aplicación sobre ella.

1.3. Estructura

Para definir apropiadamente la estructura de este proyecto vamos a dividirlo en cinco partes que no se corresponden exactamente con los capítulos del documento.

La primera parte versa sobre la explicación del proyecto, que consta de una introducción, la explicación de los objetivos y estructura.

La segunda parte contiene la revisión de las principales plataformas de aprendizaje en red que actualmente se encuentran disponibles.

La tercera parte trata acerca de las tecnologías sobre las que funciona Moodle, explicando una por una sus características. También en esta parte se incluye la revisión de las principales características de la última versión de Moodle.

La cuarta y quinta parte incluye todo lo referente al módulo dbquiz. Desde la explicación de los archivos más importantes que incluyen el módulo o las funciones más relevantes creadas hasta la documentación de un manual de usuario donde caben todos los pasos a dar para el correcto funcionamiento de este módulo.

La sexta y séptima parte trata sobre las conclusiones que se obtienen de la realización de este proyecto, así como unas líneas futuras a desarrollar para la mejora del módulo.

En la octava y última, se detalla el desglose económico que supondría el desarrollo del proyecto.

2. E-learning

2.1. [¿Qué es el e-learning?](#)

Definición según (Cruz Del Valle, 2017):

“Procesos de enseñanza-aprendizaje que se llevan a cabo a través de Internet, caracterizados por una separación física entre profesorado y estudiantes, pero con el predominio de una comunicación tanto síncrona como asíncrona, a través de la cual se lleva a cabo una interacción didáctica continuada. Además, el alumno pasa a ser el centro de la formación, al tener que auto gestionar su aprendizaje, con ayuda de tutores y compañeros.”

La idea básica por la que nace el e-learning es eliminar todas las barreras existentes entre profesorado y alumnos con el fin de acercar el conocimiento a cualquier parte del mundo.

Aunque si bien es cierto este no es el único propósito del e-learning. El e-learning permite tanto la asistencia remota como la complementación del estudio presencial. Dándole aún más calidad al aprendizaje en los centros educativos, extendiendo este fuera de las aulas y proveyéndolo de herramientas novedosas de última generación.

2.2. [Características del e-learning](#)

- El alumno es el centro de todo:

La corriente del conocimiento que sigue el e-learning es la constructivista. La cual expone que el conocimiento se crea dentro de la mente humana y que simplemente hay que proveer de herramientas para que esto se lleve a cabo. El e-learning focaliza todos los esfuerzos en esto. Dándole al alumno los medios para conseguirlo, además le permite hacerlo al ritmo que el alumno necesita para asentar los conocimientos (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013).

- Eliminación de las barreras espaciotemporales:

Como venimos explicando, el aprendizaje no queda reducido al entorno del aula y a la duración de la sesión impartida. Si no que el alumno dispone del contenido desde casi cualquier parte en la que se encuentre y las 24 horas, 7 días a la semana. Esto es un pilar básico del e-learning. Le da al alumno la libertad absoluta de articular la forma de estudio para adaptarse a sus circunstancias y necesidades.

- Mejora de los resultados:

Esto es debido a varios factores determinantes. En primer lugar, el usuario dispone de contenidos más concisos a la mano con los que poder trabajar ya que son los propios expertos, el profesorado, quien se lo proporciona. Dispone de un mayor control sobre lo que va aprendiendo ya que la evaluación es más precisa, el profesor tiene un seguimiento más exhaustivo del proceso. El aprendizaje es más inmediato, se dice que lo que se practica es lo que

realmente se aprende. El alumno dispone de entornos reales en los que ejercitarse. Esto permite una adquisición de conocimientos más rápida.

- Reducción de costos:

Es evidente que la focalización de recursos de forma más eficiente permite abaratar costes. Teniendo en cuenta que el usuario solo necesita un ordenador para conectarse a la plataforma de e-learning multitud de gastos de infraestructuras se eliminan (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013). Así como en la forma tradicional era necesario adquirir recursos de forma casi unitaria por alumno elevaba mucho los costes. Actualmente la mayor parte de los recursos son compartidos por todos los alumnos. Esto permite emplear mejor el dinero y disponer de entornos y documentación mucho más extensa y de mejor calidad.

2.3. Blended-Learning

En este punto es importante apostillar el siguiente concepto El blended-learning o b-learning consiste en la unión de la enseñanza tradicional con los métodos modernos del e-learning. Este es el formato empleado actualmente en la mayoría de los centros docentes y por extensión a los departamentos a los que va a ir dirigida la aplicación que se pretende desarrollar.

Por un lado, se sigue la metodología tradicional. Los alumnos continúan asistiendo a un lugar físico donde el docente imparte la materia en la cual el alumno recibe el grueso de la formación. Conceptos y conocimientos básicos que son imprescindibles a la hora de iniciar una andadura en cualquier formación. En paralelo, se dispone del e-learning, plataformas en las que pueden encontrar todo tipo de documentación, material de apoyo o herramientas interactivas que les permite afianzar los conocimientos impartidos o profundizar más si cabe en ellos.

2.4. ¿Qué es el LMS?

Una plataforma e-learning, plataforma educativa web o Entorno Virtual de Enseñanza y Aprendizaje es una aplicación web que integra un conjunto de herramientas para la enseñanza-aprendizaje en línea, permitiendo una enseñanza no presencial (e-learning) y/o una enseñanza mixta (b-learning), donde se combina la enseñanza en Internet con experiencias en la clase presencial (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013).

Las plataformas e-learning revolucionan la enseñanza y el aprendizaje, de forma que el proceso de aprendizaje no se queda en el aula, o espacio físico. Dando la posibilidad a alumno y profesor de establecer una relación más personalizada. Por la cual la experiencia es más satisfactoria e integral.

Las plataformas e-learning no son de uso exclusivo académico. En el ámbito empresarial son usadas a menudo por distintas corporaciones para dar a sus empleados formación y un lugar virtual donde compartir información y vivencias.

Uno de los puntos más interesantes y en cual reside el éxito de estas iniciativas es la posibilidad de tener material educativo y formas de perfección profesional accesibles permanentemente y a muy bajo coste. También cabe destacar que el uso masivo de estas tecnologías en

universidades y centros educativos está creando una potente red de usuarios y desarrolladores. La cual permite crear nuevas funcionalidades que enriquecen aún más el uso de estas plataformas. También al existir una comunidad tan extensa permite el uso más amigable para nuevos usuarios, ya que pueden resolver sus dudas y problemas en los distintos foros. Las plataformas de e-learning, que también son conocidas como Learning Management Systems, en adelante LMS, sirven como un soporte a los docentes y a los alumnos. Se considera una herramienta que permite romper con las barreras físicas, proporcionando tanto al alumno como al profesor de una experiencia más completa y enriquecedora. Los LMS permiten administrar alumnos, hacer un seguimiento exhaustivo del aprendizaje y fomentar la participación y la colaboración entre los alumnos.

Entre las principales funciones que debe tener un LMS encontramos las siguientes: Administración de usuarios, gestión de recursos y contenidos, gestión de actividades, calendario, eventos, evaluación de las actividades y administrar foros y chats.

Como características básicas que deben cumplir los LMS, según (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013), tenemos la siguientes:

- Interactividad, los LMS deben de generar la suficiente interactividad como para que los alumnos sean los que eligen como aprender.
- Flexibilidad, tanto docentes como alumnos deben de tener la posibilidad de adaptar la plataforma a sus necesidades.
- Escalabilidad, la plataforma tiene que ser capaz de llegar a tantos usuarios como se precisen.
- Estandarización, es importante que exista dicha estandarización en los LMS con el fin de proporcionar una herramienta entendible por todo el mundo y fácil de administrar.
- Usabilidad, dentro de este punto debemos contemplar el cumplimiento de 3 ítems. El primero, la efectividad, cada usuario debe poder cumplir el 100% de sus objetivos. Eficiencia, este cumplimiento de los objetivos se debe poder realizar con el menor consumo de tiempo y de recursos. Y, por último, la satisfacción de usuario.
- Funcionabilidad, las prestaciones de la plataforma deben ser las adecuadas para que la comunidad pueda desarrollar su actividad.
- Ubicuidad, todo usuario debe tener la posibilidad de conectarse a la plataforma desde cualquier lugar o dispositivo que precise y hacerlo de una forma apropiada. Es decir, el usuario debe sentirse 'omnipresente' en la plataforma.

Tipos de LMS

Los LMS podemos categorizarlos en tres tipos: Los comerciales, los LMS de software libre y aquellos que están en la nube (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013).

Los LMS comerciales son aquellos que tienen una titularidad cerrada y su fin es ser explotados comercialmente. Estos LMS tienen unas funcionalidades ya definidas por el desarrollador y un soporte técnico. En general, para hacer uso de estos LMS se compran paquetes de uso con las funcionalidades deseadas. Los puntos fuertes de estas plataformas son la fiabilidad y las funcionalidades libres de errores, ya que existen equipos de desarrollo que crean y mantienen estas plataformas. Sus puntos débiles son el coste que suponen, al ser un producto comercial requiere de licencias de uso, la comunidad de estas plataformas es relativamente pequeña comparada con los LMS no comerciales, en este caso no existe una gran comunidad en la cual

apoyarse, y por último son productos cerrados, es decir, no cabe la posibilidad de desarrollar nuevas funcionalidades. Este último inconveniente para mí es determinante ya que al no poder desarrollar la funcionalidad deseada no es de mucha utilidad. Para terminar, vamos a mencionar algunas de las plataformas comerciales más usadas: Blackboard, WebCT, OSMedia, Saba, eCollege, Fronter, SidWeb, e-educativa y Catedr@, entre otras.

LMS en la nube, no son consideradas LMS como tal. En esta categoría incluimos todas las plataformas que dan soporte a cursos MOOC o apoyo a la clase presencial. Este tipo de LMS también son cerrados y ofrecen pocas funcionalidades, la justa para que los cursos que sustentan sean viables. Es evidente que este tipo de LMS no son de nuestro interés debido a que no existe una comunidad interesada en el desarrollo y la administración de un LMS de este tipo. El objetivo es distinto, y por tanto no nos sirve en nuestra búsqueda de una plataforma LMS viable para desarrollar nuestro proyecto.

Los LMS de software libre son aquellos que son gratuitos o disponen de una licencia de tipo open source. Este tipo de LMS está desarrollado por comunidades educativas o por instituciones que no desean lucrarse con la explotación de estas. Este tipo de LMS disponen de una comunidad rica e interesada en mejorarlos. Los propios docentes buscan desarrollar distintas funcionalidades en función de las necesidades que tengan. Esto permite a estos LMS disponer de un gran repositorio de funcionalidades. Además de un conjunto importante de expertos capaces de aportar gran información y soluciones a nuevos usuarios. Esto permite tejer una gran red y hacer más fuerte al LMS. Estas plataformas tienen también sus desventajas, aunque también disponen de un equipo de desarrolladores que dan soporte y velan por el correcto funcionamiento de la plataforma, gran parte del trabajo lo desarrollan usuarios anónimos sin la supervisión de ningún equipo. Esto supone falta de estandarización, es decir, cada usuario crea en función de sus necesidades sin prestar atención a ninguna guía o manual de estandarización, lo que supone una falta de información y dificultades para desarrollar nuevos elementos, menor resistencia a los fallos, al igual que antes, los usuarios son los encargados muchas veces de generar nuevos elementos. No existe un equipo de experiencia de usuario ni una apuesta por el testing de productos. Esto puede llevar a que algunas funcionalidades sean poco robustas y tendientes a los fallos.

Este tipo de LMS son los que nos generan interés y entre los cuales buscaremos un candidato para desarrollar nuestro proyecto.

Los LMS más comunes de esta categoría son: ATutor, Dokeos, Claroline, dotLRN, Moodle, Ganesha y ILIAS.

Vamos a hacer una comparativa entre LMS de las diferentes categorías, pondremos en común los diferentes resultados y terminaremos eligiendo uno de ellos.

Aunque como hemos comentado con anterioridad los LMS comerciales no son de nuestro interés para desarrollar nuestro proyecto, vamos a ver algunos de ellos para poder tener una visión más general del mundo de los LMS con el fin de realizar una búsqueda más exhaustiva. La elección de los LMS a estudiar se da en función de la popularidad de ellos o de algunas características especiales que los hacen interesantes de revisar.

Almagesto



Ilustración 1: Portal de Almagesto

El LMS Almagesto pertenece a la compañía Alhambra-Eidos. Almagesto comienza su andadura en 1997. En un principio, se trata de una plataforma de uso interno de la empresa para la formación de sus empleados. Más tarde, otras compañías comenzaron a interesarse por esta metodología de formación. Así fue como se comenzó a comercializar este LMS.

Características:

Están tomadas de su propia web (Alhambra-eidos, 2012).

Recursos de Almagesto:

- Aula virtual de estudio
- Trabajo en grupo
- Aula de Exámenes
- Intercambio directo con los profesores
- Aula de autoevaluaciones
- Correo interno del Campus
- Listas de correo
- Sistema de envío de mensajería por el correo interno integrado a la base de datos de la gestión educativa
- Soporte administrativo
- Encuestas (Alumnos, Profesores, Cursos)
- Salas de debate
- Integración con redes sociales (Facebook, Twitter)
- Foros de debate vinculados a los cursos
- Base de conocimiento

- Conferencias emitidas en directo a través de Videoconferencias
- Videoteca
- Tablón de anuncios

Almagesto provee, de manera predeterminada, algunos roles o perfiles de usuario con distintas competencias. El nivel de acceso de cada uno de ellos puede ser modificado por los administradores, de igual modo que pueden incluirse los perfiles adicionales que se consideren oportunos. Los perfiles que se generan en la instalación de Almagesto son los siguientes:

- Coordinador General
- Coordinador de Ubicación
- Tutor
- Auditor de Ubicación
- Auditor de Alumnos
- Autor de Cursos
- Auditor de Entidad Proveedora

Servicios de e-learning:

- Servicios tecnológicos
- Ayuda en la creación y administración de contenidos
- Servicios de asesoría y consultoría e-learning
- Formación para formadores u otros perfiles en la formación virtual

Tipos de licencias:

- Alquiler Escuela
- Licencia Escuela
- Licencia Campus

Ventajas:

- Flexibilidad en la arquitectura
- Escalable
- Altamente personalizable

Desventajas (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013):

- Es una plataforma con numerosas herramientas, esto puede llevar a los tutores a emplear demasiado tiempo en el aprendizaje
- No existe una forma de solventar deficiencias y estas llegan de primera mano al usuario



Ilustración 2: Portal de Blackboard

Es el LMS comercial por excelencia, empleado en universidades de renombre, aunque también dispone de funcionalidades para centros educativos de menor grado o empresas (Inc., 2018).

Está formada por las siguientes secciones:

- Módulo de contenidos
- Herramientas de comunicación
- Herramientas de evaluación
- Herramientas de seguimiento y gestión de aprendizaje

Características:

- Experiencia de enseñanza
- Creación de comunidades de usuarios
- Gestión de los recursos
- Modelo colaborativo
- Revisión y mejoras continuas

Dispone de 4 áreas principales:

- Gestión de contenidos: Administración de información importante en el curso, documentación, elementos o materiales didácticos

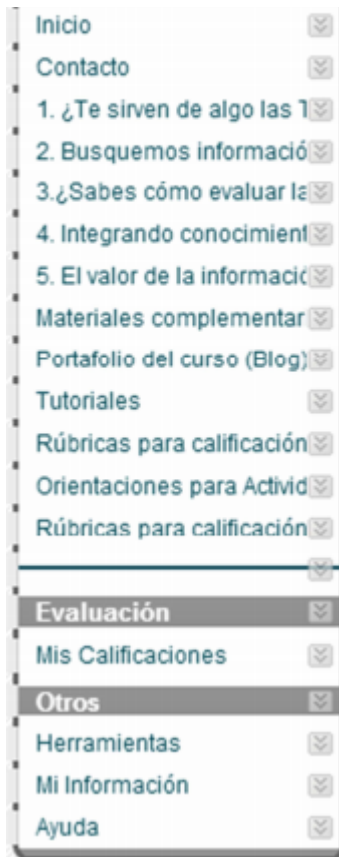


Ilustración 3: gestor de contenidos, Blackboard

En la figura podemos observar cómo se distribuyen los vínculos para poder acceder al material del curso por parte del alumno. El administrador puede gestionar estos recursos, haciéndolos disponibles o editándolos

- Comunicación: Son herramientas que permiten la interacción entre la comunidad. Ya sea síncrona o no. Existen los siguientes elementos: Foros, Chats, tableros de anuncios, transferencia de archivos, etc
- Evaluación: la plataforma tiene un amplio sistema de evaluación. Permite diseñar exámenes, encuestas o auto-exámenes. Permite la calificación y publicación de notas
- Control: El perfil de profesor tiene la posibilidad de administrar todos los recursos de un curso. Evaluación, planificación de calendario, etc

Ventajas:

- Flexibilidad: Es fácilmente adaptable a las necesidades del usuario. Así como la inclusión de otros LMS dentro de este (moodlerooms). También permite la interacción con las redes sociales

- Repositorio: Permite almacenar todo tipo de materiales didácticos, lo que facilita el manejo de recursos para mejorar la experiencia. Da una gran versatilidad a la hora de compartir este material
- Comunidades virtuales
- Extiende la colaboración más allá de la propia interacción en el aula virtual
- Esta basados sobre los principios del W3C (World Wide Web Consortium) que promueven la usabilidad, flexibilidad y la interacción
- Esta adaptado para personas invidentes

Desventajas (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013):

- No se puede usar de forma local
- Es necesario tener conocimientos informáticos, lenguaje HTML, para poder editar algunos recursos
- El rendimiento se puede ver afectado si el número de usuarios es grande
- Su interfaz no es sencilla
- Tiene deficiencias en la seguridad

e-educativa

Es una empresa de tecnología informática especializada en la implementación de proyectos y soluciones para la gestión de la formación.

Tiene una trayectoria de más de 15 años y cuentan con más de 500 clientes, más de 6000 instalaciones en todo el mundo (educativa, 2018).

Ilustración 4: Portal de e-educativa

Características:

- Interactividad: permite la interacción entre los usuarios
- Colaboración: promueve el trabajo en equipo
- Asincronismo: disponibilidad de los recursos en cualquier momento. Esto permite que el usuario tenga mayor libertad a la hora de administrar sus tareas
- Permite auto-evaluaciones
- Tiene un módulo de calendario donde cada usuario puede agendar sus eventos
- Seguimiento de las evaluaciones y los procesos
- Permite la interacción entre alumnos para promover el desarrollo colaborativo de las tareas
- Salas virtuales

Ventajas:

- Es una plataforma muy intuitiva
- No hay limitación en el número de usuarios y no se ve afectado por ello su rendimiento
- Es adaptativo a las tecnologías sobre las que corre
- Es sencillo de administrar

Desventajas (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013):

- Es limitado en el ámbito de las evaluaciones. No permite la inclusión de elementos no textuales
- Mal servicio de streaming
- No está adaptado para personas con discapacidades
- Apuesta demasiado por la asincronía lo que penaliza las herramientas interactivas

Hasta aquí hemos visto los LMS de tipo comercial, los cuales nos dan una visión bastante clara de las herramientas que existen hoy en días y de las capacidades que brindan a los usuarios. Ahora, comenzaremos a describir varias herramientas de uso libre, las cuales, como antes comentábamos, son las que verdaderamente nos interesan estudiar para decidir sobre cuál realizar nuestro proyecto.

Chamilo



Ilustración 5: Portal de Chamilo

Chamilo es una solución de software libre, licenciada bajo la GNU/GPLv3, de gestión del e-learning o aprendizaje electrónico, desarrollada con el objetivo de mejorar el acceso a la educación y el conocimiento globalmente.

Está sustentado por la Asociación Chamilo (asociación sin fines de lucro), la cual tiene como objetivo la promoción del software para la educación (y en particular de Chamilo), el mantenimiento de un canal de comunicación claro y la construcción de una red de proveedores de servicios y contribuidores al software (Chamilo, 2014)

Características (Chamilo, 2014) :

Aspectos pedagógicos:

- Se implementa de una forma que permite al profesor elegir entre una serie de metodologías pedagógicas, uno de los cuales es el constructivismo social
- El software Chamilo está construido de una manera que permite que el profesor tome el control y para que el contenido siga su verdadero lugar dentro de la experiencia del alumno
- Tiene una lista creciente de recursos de documentación disponibles en los sitios web de agregación de contenido: Youtube, SlideShare, Twitter, Vimeo, etc
- Consulte los manuales de los maestros y de administración en nuestra página de documentación
- Chamilo es usado con alumnos de 6 años a 80 años de edad, dentro de las estructuras públicas y privadas, dentro de las instituciones educativas o empresariales

Aspectos internacionales:

- Se utiliza en la producción en más de 25 países
- Es utilizado por más de 500.000 usuarios en todo el mundo
- Se desarrolla constantemente por un equipo de alrededor de 20 desarrolladores en todo el mundo
- Chamilo heredó la mayor parte de la fuerza de trabajo profesional y con experiencia de su predecesor, Dokeos
- Ha abierto traducciones para 55 idiomas, y unos 25 de ellos se encuentran más del 80 % de avance. Chino simplificado ha sido revisado recientemente por un traductor chino

Aspectos técnicos:

- Hojas de referencia rápida para los nuevos desarrolladores
- Es revisado una vez al año por un cracker de seguridad para mejorar la seguridad contra los ataques a la web. Nuevos colaboradores son entrenados durante su primera semana a enseñarles (entre otras cosas) cómo proporcionar código seguro
- Chamilo recibe alrededor de una nueva contribución importante de una nueva fuente cada dos meses. Los equipos de desarrollo actuales tienen a su cargo la revisión y la integración del código en Chamilo
- Los equipos de desarrollo de Chamilo son muy abiertos acerca de sus procesos, y siempre se puede encontrar información precisa y actualizada. Dicha información se encuentra en: <http://support.chamilo.org>
- Es de código abierto, bajo la licencia GNU/GPLv3, y está protegido por una asociación sin ánimo de lucro, lo que significa que es literalmente imposible para cualquier empresa a tomar la propiedad del proyecto y cierre hacia abajo. Pero también significa que usted es libre de desarrollar sus propias extensiones o mejoras, y compartir (o no) con la comunidad
- La asociación Chamilo ofrece un sistema de recompensa para alentar a las empresas a contribuir con código al proyecto
- Se puede combinar fácilmente con la infraestructura de TI existente. Tienen un historial probado de integración con las aplicaciones caseras, Oracle o IBM bases de datos DB2, etc

Ventajas:

- Es fácilmente usable por toda la comunidad
- Posee un diseño muy limpio y amigable
- Es de licencia de software libre
- Emplea principios pedagógicos constructivistas
- Soporta varios idiomas
- Tiene herramienta de videoconferencia
- Funciona de forma síncrona y asíncrona
- Su interfaz se puede personalizar

Desventajas (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013):

- Es tedioso instalarlo y configurarlo

Sakai

Sakai es un proyecto iniciado por las universidades de Michigan y de Indiana. Posteriormente y viendo la progresión de proyecto se unieron la universidad de Stanford, el MIT la Iniciativa de Conocimiento abierto (OKI) y el consorcio uPortal. El proyecto terminó de consolidarse con la ayuda de la fundación Mellon.

En 2004 fue la fecha de inicio del proyecto. Sakai es un entorno de código abierto. Se creó la fundación Sakai, que es actualmente la encargada de dar soporte y desarrollo a la plataforma. Actualmente a dicha fundación están adscritas más de 100 universidades.

Características (apereo Foundation, 2014) :

- Posee gran cantidad de herramientas enfocadas al desarrollo de material didáctico
- Es escalable a un gran número de usuarios
- Sus funcionalidades se pueden englobar en 4 categorías:
 - Herramientas generales de colaboración: tablones de anuncios, blog, chats, foros, calendario, etc.
 - Herramientas de enseñanza y aprendizaje: calificaciones, cuestionarios, generadores de lecciones, etc.
 - Herramientas administrativas: administración de cuentas, súper administrador, etc.
 - Herramientas de portafolios: asistentes y modelos, evaluaciones, informes, diseños y platillas de portafolios

Ventajas:

- Buenos mecanismos de gestión de la plataforma
- Esta desarrollada en JAVA lo que permite gran escalabilidad
- Al ser de código abierto permite desarrollar nuevas funcionalidades
- Posee audio conferencia
- Repositorio de archivos

Desventajas (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013):

- Tiene deficiencias en la usabilidad
- El aspecto es mejorable
- Es complejo desarrollar funcionalidades
- Al estar desarrollado en JAVA debe estar instalado en cada ordenador de los alumnos
- No tiene una documentación fiable
- Hay demasiada información, poco relevante
- La comunidad de desarrollo es bastante escasa



Ilustración 6: Portal de Moodle

Moodle, es el acrónimo de Modular Object-Oriented Dynamic Learning Environment. Es un LMS creado por Martin Dougiamas. Para ello empleó los principios pedagógicos constructivistas, que mantienen que el conocimiento se construye dentro de la cabeza del estudiante y es introducido desde fuera. Posteriormente se creó una fundación Moodle para dar soporte a la plataforma, aunque también existe una versión comercial.

Está desarrollado sobre PHP y soporta bases del tipo SQL, por ejemplo, MySQL. Se puede desplegar en máquinas UNIX y Windows dado que estas soportan PHP. Es una aplicación web gratuita que permite un amplio espectro de modos de enseñanza y perfecta para completar la enseñanza presencial. Aunque también es perfectamente válida para el tele-aprendizaje.

Es empleado en un gran número de instituciones educativas y compañías privadas.

Características (Moodle: Características de Moodle 3.3, 2017):

Características generales:

- Interfaz moderna, fácil de usar. Diseñada para ser responsiva y accesible, la interfaz de Moodle es fácil de navegar, tanto en computadoras de escritorio como en dispositivos móviles
- Tablero Personalizado. Muestra cursos de forma personalizada
- Actividades y herramientas colaborativas: foros, wikis, glosarios, actividades de base de datos, etc
- Calendario todo-en-uno. La herramienta del calendario de Moodle ayuda a mantener al día el calendario académico o el de la compañía, fechas de entrega dentro del curso, reuniones grupales y otros eventos personales
- Gestión conveniente de archivos. Administración archivos desde servicios de almacenamiento en la nube, incluyendo MS OneDrive, Dropbox y Google Drive
- Editor de texto simple e intuitivo
- Notificaciones. Cuando se habilitan, los usuarios pueden recibir alertas automáticas acerca de nuevas tareas y fechas para entregarlas, publicaciones en foros y también pueden mandarse mensajes privados entre ellos

- Monitorización del progreso. Los educadores y los educandos pueden monitorizar el progreso y el grado de finalización con un conjunto de opciones para monitoreo de actividades individuales o recursos, y también a nivel del curso

Características administrativas:

- Diseño personalizable del sitio. Fácilmente personalizable el tema de Moodle con logo, esquema de colores, etc
- Autenticación (identificación) segura e inscripciones (matriculaciones) masivas seguras. Más de 50 opciones para autenticación e inscripción, para añadir e inscribir usuarios
- Capacidad multilingüe.
- Creación masiva de cursos y fácil respaldo.
- Gestión permisos y roles de usuario
- Soporta estándares abiertos (IMS-LTI, SCORM, etc.)
- Alta inter-operatividad, integración de aplicaciones externas y contenidos
- Gestión simple de plugins
- Actualizaciones regulares de seguridad
- Reportes y bitácoras detalladas

Características para Desarrollo y Gestión del curso

- Rutas directas de aprendizaje. Las clases pueden ser dirigidas por el instructor, auto-reguladas, mixtas o completamente en-línea
- Fomento de la colaboración
- Inclusión de recursos externos
- Integración Multimedia
- Gestión de grupo
- Flujograma de puntuación
- Calificación en-línea
- Evaluación propia y por pares. Actividades incluidas, tales como talleres y encuestas, estimulan a los alumnos para que vean, califiquen y evalúen el trabajo de ellos mismos y el de otros participantes del curso como un grupo. Insignias integradas. Totalmente compatible con las Insignias Abiertas de Mozilla (Mozilla Open Badges)
- Resultados y rúbricas. Métodos avanzados de calificación para personalizar el libro de calificaciones del curso
- Puntuación basada en competencias
- Seguridad y privacidad

Ventajas:

- El docente controla todo el contenido del curso
- El profesor puede monitorizar el desarrollo de actividades
- Existe la posibilidad de fijar plazos de entrega
- Se puede reutilizar estructura y contenidos de los cursos
- Posibilidad de compartir recursos
- Posibilidad de múltiple gestión de los cursos por parte de varios profesores
- Permite la inclusión de recursos de muchos tipos: Texto, audio, video, etc
- Existen flujos sencillos de comunicación entre alumnos y docentes
- La evaluación es muy activa y dinámica, el alumno tiene un feedback casi permanente

- Es altamente personalizable
- Permite al profesor ver el curso como si fuese un alumno
- Está disponible en más de 70 idiomas
- Es flexible con los ritmos de trabajo
- Los modelos de examen en la plataforma son muy interesantes y permiten un feedback inmediato
- Dispone de una gran comunidad de desarrollo
- Dispone de documentación bien organizada en función de la versión de Moodle

Desventajas (Mariel Castro, Ariel Clarenc, López de Lenz, Moreno, & Tosco, 2013):

- Deja algunos aspectos de la seguridad al host donde se aloja, lo cual no es recomendable
- No dispone de aplicación de videoconferencias
- Si la plataforma se despliega para muchos usuarios hay que ser muy cuidadoso con la configuración del servidor
- No permite gestión económica de los alumnos

Una vez hecha esta revisión de distintas plataformas LMS se llega a la conclusión de que la más idónea para acoger el desarrollo del proyecto es Moodle. A parte de ser el LMS que se emplea en la universidad a la que pertenecemos, tiene varios puntos fuertes que nos hacen decantarnos por ella. Entre ellos mencionaremos los siguientes:

- Dispone de una amplia comunidad de desarrolladores y foros de desarrollo en los que plantear tus dudas
- Dispone de documentación bien organizada y disponible en su web
- Los propios foros son una fuente muy fiable de información
- Es un LMS web lo cual nos permite hacer uso de HTML, CSS y javascript. Además, está desarrollado en PHP y MySQL. Todos estos son lenguajes estudiados y conocidos en nuestro grado

3. Moodle 3

Como anteriormente ya se determinó la idoneidad de Moodle para el desarrollo de este proyecto, vamos a ver más de detalle las características de la última versión de Moodle sobre la que se ha desarrollado la aplicación DBquiz. En el anterior apartado también se explicó la base tecnológica sobre la que está desarrollado Moodle por tanto ahora vamos a explicar a más alto nivel que supone para el usuario final Moodle. Vamos a enfocar las características de Moodle en función de usuario, ya sea administradores, profesores o alumnos.

3.1. Perfil de administrador

El perfil de administrador es el que toma el control del sitio en cuestión inmediatamente después de su creación, aunque posteriormente puede delegar algunas funciones en otros usuarios con el rol de Manager asignado (Moodle: Gestionando un sitio Moodle, 2017).

Características como administrador:

La explicación de las características se agrupará en función del ámbito al que pertenecen para conseguir dar una visión más global de las funciones del administrador.

-Usuarios y Cursos:

Métodos de Autenticación:

Existen tres formas de autenticar usuarios en Moodle que son.: Ingreso manual por parte del administrador, autenticación a través de auto-registro basado en Email y autenticación contra servidores de autenticación.

-Ingreso manual:

En este caso todo el proceso depende únicamente del administrador del sitio. Desde la plataforma, el administrador deberá introducir uno a uno los usuarios rellenando un formulario con los datos del usuario. Posteriormente deberá ceder las credenciales al usuario en cuestión. El administrador puede forzar, si lo desea, la petición de cambio de contraseña en el primer acceso al sitio por parte del usuario. En el momento que el administrador genera este tipo de usuario queda verificado y listo para usarse.

-Autenticación a través de auto-registro basado en mail:

Es necesario que el administrador del sitio habilite esta opción. No es una forma de autenticación muy recomendada ya que el administrador pierde el control de quien se registra. Lo cual puede dar situaciones indeseadas, o fallas en la seguridad del sitio el cual puede llegar a verse comprometido. Lo ideal para este caso es habilitar la opción por un periodo corto de tiempo controlado.

Para el control de autenticación se puede habilitar un complemento de reCAPTCHA que permite al sitio Moodle en cuestión identificar comportamientos sospechosos para evitar la entrada de botnet que puede causar problemas de fraude y spam.

Para acceder el usuario necesitará cumplimentar el formulario de autenticación y proporcionar una cuenta de correo válida, ya que al terminar el sitio enviará un mail de verificación a esta cuenta. Si el usuario no verifica la cuenta vía mail. Al cabo de un tiempo se suspenderá el usuario.

-Autenticación contra servidores de autenticación:

El principio de funcionamiento de este modo de autenticación es una forma muy popular para enrolar usuarios en distintas plataformas. Debido a que es una forma muy segura de realizar dicha autenticación, además permite la migración de usuario de forma rápida.

Para autenticar usuario contra un servidor es necesario que por parte del administrador tomar ciertas acciones con respecto al servidor de autenticación. Es necesario el intercambio de certificados de seguridad por parte de ambos. Esto permitirá el cifrado de las conexiones para un intercambio seguro de información. Por otro lado, también será necesario una correcta configuración del servidor de autenticación, la cual puede recaer o no sobre la misma persona que administra el sitio Moodle.

Existen multitud de servidores de autenticación y de protocolos de autenticación. Algunos son los siguientes:

- LDAP
- IMAP
- Servidor CAS
- FirstClass
- NNTP
- POP3 Server
- Radius Server
- Shibboleth

Gestión de cuentas:

El administrador tiene la capacidad de administrar el resto de cuentas de usuarios con el fin de proveer de información y un correcto funcionamiento del sitio. Puede listar los usuarios y dentro de estas listas realizar tareas de búsqueda, filtrado y edición de cuentas. Puede tomar acciones masivas con los usuarios. Puede realizar envío masivo de mail, o tareas de forma global como: Confirmar cuentas, eliminar cuentas, exportar datos de usuarios o forzar cambios de contraseña.

Todo esto se lleva a cabo de la misma forma, desde la lista de usuarios podrá seleccionar a los que desee y desde ahí tomar las acciones oportunas.

También puede realizar tareas como la creación de usuarios, anteriormente vista, modificación de los datos de un usuario o cargar imágenes de usuarios.

Roles y permisos:

En Moodle cada usuario dispone de un rol. Los roles están predeterminados y permiten acceder a los usuarios a determinadas funciones. Es decir, dependiendo del rol que posea un usuario tendrá mayores o menores privilegios dentro el sitio.

Es labor del administrador gestionar el rol asociado a cada usuario. El administrador puede añadir, modificar o eliminar el rol de un usuario.

Asimismo, tiene la posibilidad de añadir, modificar o eliminar roles. Es decir, puede modificar los permisos dentro de un rol.

Matriculaciones a cursos:

Aunque los usuarios puedan acceder al sitio Moodle porque estén autenticados, no quiere decir que tengan acceso a todos los cursos que haya en el sitio. Deben estar matriculados para poder hacer uso del curso. Existen distintas formas de matricular usuarios a un curso. Vamos a repasar las distintas formas.

Inscripción manual: El administrador del sitio y también el profesor con privilegios de gestión de cursos pueden incluir manualmente a todos los alumnos que estén registrados en el sitio que deseen.

-Auto inscripción:

El administrador del sitio o profesor con privilegios puede elegir esta opción. Para restringir el acceso a usuarios no deseados también se puede crear una contraseña. De tal forma que al ingresar dicha contraseña el alumno quedará matriculado en el curso. También se puede habilitar esta opción solo por un periodo de tiempo limitado o crea una cota máxima de inscritos.

-Sincronización de cohortes:

Las cohortes son grupos a los que un usuario puede pertenecer. Si el administrador puede asociar una cohorte a un curso y automáticamente los usuarios de esta cohorte quedarían matriculados al curso.

-Enlace a un meta curso:

Se pueden asociar cursos de tal forma que todo usuario que ya estuviese matriculado a un curso quedaría matriculado del segundo.

-Inscripciones por categoría:

El administrador puede incluir una categoría en un curso de tal forma que todos los usuarios que dispongan de dicha categoría quedarían inscritos al curso.

-Inscripciones contra una base de datos externa:

De igual forma que anteriormente vimos cómo se puede autenticar a un usuario contra un servidor externo podremos matricular a un usuario en un curso. Simplemente configurando correctamente el servidor y haciendo uso de certificados de seguridad los usuarios incluidos en dichos servidores serán matriculados en el curso.

-Volcados de usuarios a través de ficheros:

El administrador podrá hacer una importación de usuario desde un fichero. Moodle da la opción de que a través de un csv debidamente organizado podamos incluir todos los alumnos que deseemos en un curso. De igual forma, está permitido hacer la matriculación de usuario a través de un fichero XML.

-Acceso mediante perfil de invitado:

El administrador tiene la opción de permitir el acceso a cualquier usuario en forma de invitado de tal forma que este podrá acceder al curso y visualizar el contenido sin poder participar en el curso directamente.

-Tareas de servicio y seguridad

Tareas de servicio:

Se trata de un conjunto de herramientas con las que el administrador puede desarrollar tareas de revisión del servicio para poder comprobar el correcto funcionamiento del sitio y poder tomar las medidas oportunas en caso de exista algún problema en el sitio Moodle.

A continuación, se enumeran las herramientas disponibles para ejercer las tareas de servicio:

-Dev/Prueba de aceptación: Permite al administrador mostrar errores y advertencias en la ejecución de la base de datos y el php, esto permite desarrollar nuevos módulos y hacer pruebas en el sitio Moodle.

-Herramientas de pruebas de salud: Esta herramienta permite monitorizar el estado del sitio, reporta latencias estado de la máquina, etc. Permite medir la salud del sitio.

-Actualización: Permite comprobar si existen actualizaciones de Moodle o de los módulos instalados y si es así descargar y actualizar. También esta herramienta permite instalar nuevos plugins y paquetes de idiomas.

-Gestión de base de datos: Permiten al administrador multitud de funciones a la hora de gestionar la base de datos del sitio. Desde buscar y modificar campos en la base de datos a hacer migraciones.

Seguridad:

Moodle provee herramientas para comprobar si el sitio es seguro y a su vez introduce medidas de contención en caso de fallo.

-Respaldo: Permite realizar copias de seguridad y programarlas para su realización generando así un respaldo en caso de ser necesario recuperar el sitio ante una falla de seguridad o una caída.

-Recomendaciones de seguridad: Proporciona un diccionario de recomendaciones y buenas prácticas para mantener seguro el sitio.

-Reporte de Seguridad: Hace una revisión de riesgos potenciales conocidos para subsanar posibles deficiencias encontradas.

-Configuración de políticas y privacidad: Permite configurar políticas y restringir la privacidad tanto como el administrador desee.

-Configuraciones para todo el sitio:

Idioma:

El administrador tiene la posibilidad de modificar el idioma del sitio Moodle. Si se desea se pueden descargar paquetes de idiomas adicionales o personalizar los ya existentes modificando o incluyendo nuevas palabras en el diccionario de Moodle.

Ubicación:

Es posible configurar las zonas horarias dentro del sitio con el fin de adecuar las fechas de realización de actividades en los cursos.

3.2. Perfil de profesor

Este apartado vamos a describir las características más relevantes de rol del profesor (Moodle: *Managing a Moodle course*, 2017). Si bien cabe destacar que el rol de profesor debe dividirse en dos. En primer lugar, está el rol de profesor con privilegios, el cual tiene permisos para administrar los cursos en los que este asignado, la mayoría de las características que detallaremos pertenecen a este rol, pues el otro rol de profesor es el profesor sin privilegios que limita su actividad a la de ser un calificador más de tareas. Este nunca podrá realizar cambios en las actividades ni podrá crear nuevos recursos en el curso.

Como ya hemos comentado, la tarea del profesor se focaliza básicamente en la gestión del curso. El curso como tal pretende ser un espacio personalizado por el profesor para abarcar todas las necesidades de este. Por tanto, puede ser un simple repositorio de recursos en los que apoyarse a la hora de impartir docencia o bien un completo curso online, con recursos y actividades organizados en los bloques que se deseen.

Referente al curso propiamente el profesor podrá, crear el curso, organizar categorías dentro del curso, controlar la forma en la que cada usuario en función de su rol, categoría, etc., ve el curso, darle un formato determinado al curso o restringir accesos al curso.

Dentro de la comunidad de Moodle existen infinidad de proyectos que permiten al profesor desarrollar un gran potencial dentro del curso, siempre que lo desee mediante el administrador puede instalar estos plugins y generar multitud de actividades dentro del curso. También existen por defecto gran cantidad de plugins pertenecientes a Moodle que permiten entre otras cosas, hacer encuestas, examinar a los alumnos con tests o respuestas cortas, añadir documentos o editar html para construir dentro del propio curso frames.

3.3. Perfil de alumno

Dentro del ámbito de los alumnos, Moodle ha sido optimizado para que el alumno encuentre todo tipo de herramientas útiles para su estudio. Moodle ha ido perfeccionando aspectos esenciales que permiten al alumno desarrollar su formación si echar en falta prácticamente ninguna herramienta.

Dentro de Moodle el alumno dispondrá de mensajería instantánea tanto en grupos como individual con otros alumnos o profesores, repositorios privados para almacenar documentos personales, gestión de su propio calendario. Esto supone un avance a la hora que el propio alumno sea dueño de su tiempo y realice sus tareas cuando prefiera. También con el avance de las tecnologías, se ha hecho hincapié en la adaptación de Moodle a los nuevos dispositivos. Ahora Moodle dispone de los diseños adaptativos más vanguardistas.

También se dispone de una app para dispositivos que dispone de las mismas funcionalidades que la versión web y además permite establecer notificaciones para llevar al día todo lo que sucede en el curso (Moodle: *Moodle Mobile features*, 2017).

3.4. Consideraciones para desarrollar en Moodle

En este apartado vamos definir varios puntos a tener en cuenta a la hora de iniciarnos en el desarrollo de funcionalidades en Moodle.

En primer lugar, Moodle tiene establecido un protocolo básico para desarrollar módulos. Desde su página web (Moodle: NEWMODULE Documentation, 2017). Existe una guía básica desarrollada por Moodle donde, descargando un paquete con templates y siguiendo unos sencillos pasos dispondremos de nuestra primera versión del módulo. Es muy básico, pero contiene los archivos necesarios para comenzar. Estos archivos pueden ser customizados como se desee con el fin de darles las funcionalidades oportunas o bien crear nuevos que complementen a estos.

Para continuar, Moodle funciona sobre distintas tecnologías ampliamente usadas y conocidas, pero en la realidad Moodle desarrolla y funciona sobre sus propias librerías de código. Para que el trabajo que vamos a desarrollar sea efectivo y perdure en el tiempo es necesario que sea desarrollado empleando dichas librerías. Todas las funcionalidades deben ser en su completitud construidas sobre las funciones desarrolladas por Moodle. De esta forma nos aseguramos que nuestro trabajo vaya de la mano con las actualizaciones de la plataforma. De esta forma, las actualizaciones sobre dichas funciones, cambios de tecnologías, actualización en las versiones, etc. no afectarán a nuestro trabajo. Un ejemplo claro es el acceso a memoria por parte de nuestros plugins. Moodle dispone de sus propias funciones, construidas desde primeras versiones de la plataforma. Al emplear estas funciones, evitamos que nuestro módulo sufra con las actualizaciones de la plataforma, ya que desde que se crearon dichas funciones, SQL se ha actualizado y ha aparecido, por ejemplo, SQL orientado a objetos.

Por otro lado, a nivel de seguridad es importante tener en consideración los accesos a memoria. El propio módulo posee, por exigencias de Moodle, su propia base de datos independiente. Esto previene los accesos al core de la memoria de Moodle, impidiendo que se hagan modificaciones en la información estructural del sitio. Aunque si bien es cierto que se pueden realizar accesos a la base de datos que contiene información de los usuarios, es preferible hacerlo de la forma más controlada posible, intentando acotar al máximo estos accesos.

En cuanto a la apariencia del módulo, es necesario tener en cuenta que, Moodle dispone de sus propios temas. Moodle permite modificar la apariencia del sitio. Pero si nuestro interés es darle una apariencia distintiva a nuestro módulo deberemos tener en cuenta diversos puntos:

- Para empezar, es necesario crear archivos CSS para el módulo. Estos archivos deben ser incluidos en un lugar determinado del módulo para que sean correctamente leídos.
- Como buenas prácticas, se entiende que el uso de estilos se hace desde archivos específicos para ello. Evitando, por tanto, incluirlo en el propio código del módulo.
- A la hora de emplear los archivos de CSS es necesario entender que se debe hacer como herencias de los archivos CSS globales, es decir, todos los cambios producidos en el estilo deben producirse sobre etiquetas ya incluidas en las hojas de estilo del propio sitio Moodle. Esto nos permitirá mantener el diseño responsive y que no haya errores en la presentación del módulo.

3.5. Tecnologías incluidas en Moodle

En este apartado vamos a hacer una revisión de todas las tecnologías empleadas para el desarrollo del módulo dentro de Moodle.

Como ya sabemos Moodle está desarrollado en PHP y JavaScript. Aunque como está implementado en web, también es necesario tener en cuenta el uso de HTML5 y CSS. Así como el lenguaje de base de datos MySQL.

La instalación de la base de datos se hace con el editor XMLDB, es una herramienta de xml que especifica como Moodle debe configurar sus tablas de la base de datos (Moodle, 2017).

3.5.1. PHP

PHP fue creado en 1994 por Rasmus lerdorf (The PHP Group, 2017). Originalmente se desarrolló en C para rastrear visitas a una página en la que Rasmus tenía su CV. Lo llamo Personal Home Page Tools, PHP tools. Posteriormente, y buscando mayores funcionalidades, reescribió el código para darle mayores y mejores funcionalidades. Ya disponía funciones como la interacción con bases de datos y creación de webs dinámicas. En 1995, Rasmus publicó el código fuente, lo que fomentó la incorporación de nuevas soluciones y solución de errores por parte de nuevos usuarios.

Poco después PHP pasó a llamarse FI, forms Interpreter. En este punto ya disponía de funcionalidades básicas que hoy conocemos con la interpretación de formularios y la sintaxis HTML incrustada.

En 1996, se volvió a reescribir el código al completo y pasó a llamarse PHP/FI. En este punto ya se podía considerar PHP un lenguaje de programación. En este punto ya incluía soporte de bases de datos, SQL, postgre95, cookies, etc.

Entre 1997 y 1998 contaba con una gran popularidad, de hecho, era el lenguaje preferido en programación web.

PHP 3 apareció como sustituto de PHP/FI. Comenzó a desarrollarse debido a la falta de desarrollo e ineficiencia de PHP/FI para ciertos proyectos. Andi Gutmans y Zeev Suraski, junto a Rasmus Lerdorf, comenzaron un nuevo proyecto que supondría una completa revolución.

PHP 3.0 gozaba de una gran extensibilidad y proveía de una interfaz madura para múltiples bases de datos, protocolos y APIs. La sencillez de ampliación de lenguaje hizo que se formase una gran comunidad de desarrolladores y permitió aunar fuerzas para ampliar las funcionalidades del lenguaje.

PHP 4 comenzó a desarrollarse poco después de ser publicado PHP 3. Andi Gutmans y Zeev Suraski, comenzaron a desarrollar un nuevo núcleo de PHP. La idea era mejorar la ejecución de aplicaciones complejas y la modularidad del código base.

Esto permitió la incorporación de infinidad de funcionalidades, soporte en la mayoría de servidores web, sesiones HTTP, etc.

PHP 5 contiene una mejora del núcleo de PHP 4. Se lanzó después de un largo desarrollo y varias versiones. Sus principales mejoras son un nuevo modelo de objetos y nuevas opciones.

En la actualidad existe la versión 7. No existe la versión 6, ya que fue un proyecto fallido y cuando se inició se prefirió no reutilizar el nombre.

Funcionalidades de PHP:

En esencia se emplea en el desarrollo de scripts en servidores, se usa para administrar formularios, generar contenidos dinámicos o gestionar cookies (The PHP Group, 2017).

PHP se extiende en tres campos donde se pueden implementar sus scripts.

En primer lugar, veremos el lado de servidor, el cual es el principal. Dentro del servidor web incluiremos los archivos necesarios con el código php. Estos archivos serán interpretados por el analizador de PHP. Una vez realizado, se enviará al navegador la página construida.

Otra forma de implementar scripts es desde la línea de comandos. No es necesario tener un servidor ni un navegador. Solo se necesita un analizador de PHP. Estos scripts pueden usarse en tareas simples de procesador de textos.

Por último, se emplea en aplicaciones de escritorio. No es el método más ortodoxo para hacerlo. Pero si se dispone de los suficientes conocimientos y se pretende implementar alguna funcionalidad de las que dispone PHP se puede hacer perfectamente usando PHP-GTK. Es una extensión de PHP que permite este tipo de aplicación. Aunque no es disponible en la línea principal del producto.

Es un error pensar que PHP simplemente limita a generar HTML. Es capaz de generar imágenes, PDF o películas flash.

Un punto fuerte de PHP es el soporte de base de datos que posee. Existe un sinnúmero de bases de datos soportadas por PHP, así como funciones de gestión de estas realmente sencillas.

También da soporte a una gran cantidad de protocolos, así como crear sockets.

3.5.2. JavaScript

JavaScript nació al igual que PHP para dar solución al problema de las páginas web del momento, estáticas y que no permitían interacción con el usuario final. HTML solo permite mostrar textos planos con estilos (Valdés, 2007).

JavaScript comenzó a desarrollarse por Netscape en 1995 con el nombre de MOCHA. Al poco tiempo pasó a denominarse LiveScript. En diciembre de 1995 se unió al desarrollo de JavaScript Sun Microsystems, creador de Java. En ese momento pasó a llamarse JavaScript debido a que en ese momento Java gozaba de una gran popularidad y al adoptar este nombre y comenzó a generar interés en la comunidad de desarrolladores.

En ese momento, Microsoft desarrolla JScript que lo introduce en Explorer 3.0. Se trata de un lenguaje muy parecido a JavaScript. Con ello pretende entrar en la carrera de desarrollo de lenguaje para la dinamización de páginas webs.

En 1996 JavaScript se estandariza en conjunto con ECMA y W3C. Se crea el DOM, document object model. Este fue muy importante debido a que se podía gestionar dinámicamente el documento web de una forma rápida y eficiente (jlopezfdez, 2017).

JavaScript en la actualidad es el lenguaje base de multitud de frameworks y librerías, jquery, zepto, angularJS, etc., así como del motor web node.js. Esto supone una revolución a la hora de generar plataformas web y nuevas aplicación suponiendo verdaderamente una revolución en el

desarrollo de apps y webs ya que la velocidad de procesado y la dinamización que tiene no se ha conseguido replicar por otros lenguajes o plataforma.

Características de JavaScript:

Javascript es un lenguaje no tipado. Esto quiere decir que la declaración de variables no exige un tipo de datos en concreto. Las variables toman el tipo de datos cuando se almacena.

JavaScript posee dos vertientes. La primera, la del lado cliente y la otra es la del lado de servidor.

JavaScript se puede ejecutar en el lado cliente. Se usa para controlar el navegador y el DOM. Permite modificar la vista web sin necesidad de hacer consulta al servidor. Lo que proporciona mayor versatilidad y menor tiempo de espera. Se trata de script, normalmente de pequeño tamaño que se encargan de realizar funciones muy concretas.

También se puede ejecutar en el lado servidor. En ese sentido entra en juego el anteriormente mencionado node.js. Se trata de un motor de ejecución de servidor que permite la comunicación con la base de datos. En este sentido se abre un abanico de posibilidades muy interesante a todos los niveles. Existen multitud de frameworks de JavaScript muy punteros para el desarrollo de plataformas web o aplicaciones. En este caso es necesario mencionar JQuery, Zepto, AngularJS o Bower. Estos frameworks están desarrollados por grandes compañías como Google en el caso de Angular. Esto proporciona una idea del nivel de dichos frameworks y sus posibles utilidades.

3.5.3. CSS

CSS son las siglas de Cascade Style Sheets, hojas de estilo en cascada. El uso de este tipo de tecnología es la que permite darle cierto aspecto o apariencia a un documento en HTML. CSS está inequívocamente ligado al lenguaje HTML (Schools, 2018).

La primera vez que se empleó CSS fue en 1970, posterior a la creación del lenguaje SGML. Se vio claramente la necesidad de crea un estándar para dar formato a los distintos documentos electrónicos.

En un momento de gran crecimiento de internet y con la aparición de HTML, aparecieron numerosos navegadores, esto generó un problema de incompatibilidades y gran dificultad a la hora de dar apariencia a los documentos en internet. El consorcio W3C creado una proposición de estandarización y de ahí en 1995 nació el CSS tal y como lo conocemos hoy.

En 1995 W3C publica oficialmente su recomendación de uso de hojas de estilo, que se conoce como CSS de nivel 1.

Posteriormente, con la aparición de nuevos elementos y evolución de nuevos estándares de HTML y la modernización de la navegación web, pareció CSS 2, en 1998. Poco después se inició una revisión de CSS 2, conocida como CSS 2.1 que fue la que finalmente se implantó en los navegadores. Corregía algunos fallos y eliminaba funcionalidades no usadas (Anónimo, Universitat Oberta de Catalunya, 2017).

En la actualidad existe CSS 3 que, aunque continua en desarrollo, supuso un gran avance al ser modular y disponer de gran cantidad de nuevas funcionalidades.

CSS funciona creando normas que le indican al navegador como crear un elemento exactamente. Estas normas pueden dar formato a un tipo elemento en concreto del documento,

a una 'clase' de elementos, que son englobados con el atributo "class" en el HTML o a un elemento en particular a través de su id.

Este formato va desde el ancho en pixeles, al color y son los responsables finales de como se muestra el documento.

Las ventajas de que tiene usar un archivo CSS en programación web son numerosas. En primer lugar, separar el estilo del HTML permite una programación mucho más limpia y ordenada. De otra forma el código queda muy enmarañado y difícil de modificar en caso de ser necesario.

Otra de ellas es que, a la hora de diseñar, es mucho más fácil generar nuevos aspectos o modificar los existentes.

CSS en la actualidad permite, en función del dispositivo sobre el que se represente la información y sus características, adaptar sus estilos. Esto se realiza mediante media queries como bootstrap. Esto se conoce como diseño adaptativo o RWD.

El estilo nos permite organizar la información dentro de nuestra página, permitiéndonos colocar el contenido como deseemos y no a la libre interpretación del navegador de turno.

Al estar estandarizado la gran mayoría de navegadores lo utilizan y no necesitaremos definir estilos para cada navegador.

3.5.4. HTML

HyperText Markup Language es el nombre completo de HTML. Es un lenguaje de programación basado en etiquetas que permite definir los distintos elementos en un texto para ser publicados. Esto le da una estructura y forma deseada al documento. Es utilizado esencialmente en el diseño de páginas web (Anónimo, Librosweb, 2017).

HTML es ideado en 1980 por Tim Berners Lee, empleado del CERN, que lo propuso como sistema de "Hipertexto" para compartir documentos. Esta idea no era nueva ya que ya existía como tal, SGML es un ejemplo. HTML lo que venía a solventar era la dificultad para manejarse con los extensísimos lenguajes de hipertexto del momento. Estos documentos esencialmente enlazaban unos documentos con otros de tal forma que desde uno se podía acceder a otros. Básicamente como se hace en la actualidad con los hipervínculos.

El primer documento formal de HTML data de 1991 conocido como HTML Tags.

En primer lugar, se hizo cargo de desarrollar dicho lenguaje la IETF, Internet Engineering Task Force. Esta primera propuesta de estandarización se presentó en 1993, aunque realmente no llegaron a formalizarse como estándar.

En 1995 se crea dentro de IETF un equipo de trabajo para desarrollar HTML. Se consigue publicar HTML 2.0, el cuál es su nombre oficial, aunque se trate realmente del primer estándar de HTML.

A partir de 1996 toma las riendas del proyecto W3C y en 1997 se presenta HTML 3.2 que incorpora multitud de avances, como applets de java.

HTML 4.0 hace aparición en 1998 e incorpora las hojas de estilo, toda una revolución, inclusión de scripts, etc. Posteriormente se publicó la versión 4.1 que corregía algunos problemas.

En este punto W3C pierde el interés en seguir desarrollando HTML y se centra en XHTML. Por esta razón, aparece la asociación WHATWG(Web Hypertext Application Technology Working

Group) respaldada por las grandes empresas desarrolladoras de navegadores, dado que estas empresas no estaban dispuestas a cambiar sus desarrollos para adaptarse a un nuevo estándar.

HTML5 vio la luz en octubre de 2014. Esta nueva versión supone un avance radical en la forma de entender el desarrollo web ya que incorpora infinidad de nuevos elementos y funcionalidades (alfredotemiquel, 2017).

Vamos a comentar brevemente las funcionalidades de las que dispone HTML5.

Semántica:

- Nuevas secciones y elementos, <section>, <article>, <nav>, <header>, <footer>, <aside>, etc.
- Formularios: Mejora de la API, nuevos atributos, etc.
- Mejora de iframe: Nuevos atributos y mejoras en la seguridad
- MathML, permite integrar fórmulas matemáticas

Conectividad:

- Web sockets, permite crear conexiones permanentes entre servidor y cliente, así como el intercambio de datos no HTML.
- Eventos de servidor, permite colocar eventos en el lado cliente.
- WebRTC, comunicación en tiempo real

Almacenamiento y recursos sin conexión:

- Permite operar con recursos de la cache de la aplicación.
- Soporta eventos con y sin conexión
- Soporta IndexedDB

Multimedia:

- Elementos específicos de audio y video: <audio> y <video>.
- Elemento <canvas> que permite dibujar gráficos
- SVG, formato de imágenes vectoriales para canvas
- Uso de la API de la cámara del dispositivo
- Eventos táctiles
- Geolocalización

3.5.5. SQL

SQL (Structured Query Language) es un lenguaje de gestión de base de datos relacionales. SQL tiene sus inicios en la compañía IBM. Esta creó, en la década de los 70, un nuevo software de base de datos llamado System R. Para el lenguaje encargado de gestionar los datos almacenados se creó SQL.

Posteriormente, en 1979, la compañía Relational Software, que después se convertiría en ORACLE, lanzó la versión 2 de SQL. Que se trataba de la primera versión comercial.

En 1986, por primera vez se tiene un estándar de SQL por ANSI (American National Standards Institute), nombrado SQL-86 o SQL1.

En 1992, se lanza un nuevo estándar de SQL, SQL-92 o SQL2. Este estándar viene a cubrir deficiencias existentes y ciertas necesidades que tenían los desarrolladores.

La siguiente versión del estándar aparece en 1999 con el nombre SQL2000. Se comienza a dar al lenguaje ciertas características orientadas a objetos. Se incluyen expresiones regulares y consultas recursos para las relaciones jerárquicas.

Durante los sucesivos años, han continuado apareciendo nuevas versiones o mejoras que han incluido nuevas expresiones, características ligadas a XML, como la importación de datos a este formato, lenguaje de consulta XML, etc.

En 2016, la última versión hasta la fecha, comienza a integrarse con el lenguaje JSON, muy empleado en los últimos tiempos por los desarrolladores.

Características (Groff, Weinberg, & Opper, 2010):

- Lenguaje de definición de datos (DDL): Se trata de comandos para la creación, modificación o borrado de esquemas de relación. Estas órdenes permiten introducir, modificar o borrar tablas en las que, posteriormente, se almacenarán los datos.

Algunas de las órdenes son:

- CREATE: Creación de tablas
- DROP: Eliminación de tablas
- ALTER: Modificación de la estructura de las tablas
- TRUNCATE: Este comando elimina todo el contenido de la tabla

- Lenguaje de manipulación de datos (DML): Este lenguaje está constituido por las distintas consultas empleadas para la manipulación de los datos incluidos en las tablas anteriormente mencionadas.

Algunas de las órdenes son:

- SELECT: Permite seleccionar datos incluidos en las tablas. La utilización de la cláusula WHERE permite filtrar la búsqueda de datos en la tabla.
- INSERT: Se emplea en la introducción de nuevos datos en las tablas. También permite el uso de cláusula WHERE.
- UPDATE: Modifica la información indicada en la tabla deseada.
- DELETE: Elimina la información indicada.

4. Módulo dbquiz

4.1. Introducción

Para cumplir con los requerimientos propuestos, se va a desarrollar un módulo dentro de la plataforma de Moodle que, por un lado, permitirá al profesor crear una base de datos con un script sobre la que después podrá añadir preguntas para realizar consultas. Incluir preguntas, sobre cómo realizar consultas a la base de datos.

Posteriormente, deberá permitir al alumno hacer frente a la resolución del ejercicio. Por un lado, la parte de la creación de la base de datos y por otro, la proposición de consultas a la base de datos para devolver los campos solicitados en la pregunta.

Para finalizar, el profesor deberá poder dar una calificación al alumno en base a sus respuestas.

4.2. Requisitos funcionales

Este módulo tiene como finalidad crear ejercicios interactivos en los cuales el alumno pueda entrenar sus capacidades en el manejo de las bases de datos. Por un lado, el módulo permite experimentar con la creación de bases de datos y por otro, permite aprender cómo hacer consultas a una base de datos de forma apropiada.

4.2.1. Requisitos básicos

-Permitir al profesor incluir un script de creación de base de datos en MySQL y que este almacene preguntas para el posterior ejercicio de consulta a la base de datos. También deberá permitir la inclusión de una imagen referente a la base de datos para poder desarrollar la segunda parte del ejercicio correctamente. El módulo deberá permitir incluir distintos elementos propios de la gestión del ejercicio, como el enunciado, fechas de apertura y cierre del ejercicio, número de intentos, si se trata de un examen, etc.

-Permitir almacenar por parte del alumno las respuestas proporcionadas tanto para la base de datos propuesta como de las respuestas a las preguntas de las consultas. Aunque si bien es cierto que sería atractivo que el alumno pudiese tener un feedback momentáneo de su consulta, es una falla de seguridad importante. Aun poniendo cotas a las consultas, una persona con suficientes conocimientos podría atacar el sitio Moodle y poner en serio riesgo su integridad.

-Evaluación por parte del profesor. El módulo deberá permitir a profesor ver las respuestas completas del alumno y proveer un sistema de evaluación para que este califique al alumno.

-Visualización de la calificación. Una vez calificado el ejercicio, el alumno deberá poder ver la calificación otorgada a su ejercicio y los comentarios del profesor.

4.2.2. Requisitos técnicos

Como ya se ha explicado anteriormente, Moodle opera sobre unas tecnologías específicas y que por tanto todo lo que se desarrolle debe ser conforme a las normas que estas imponen. Moodle está desarrollado sobre PHP, MySQL y JavaScript entre otras cosas.

Por tanto, los siguientes puntos de requisitos atienden en mayor o menor medida a ellas:

-Como es obvio las sentencias del lenguaje escrito deben ser en PHP, o script de JavaScript. Las conexiones a la base de datos deben ser con peticiones de MySQL.

-Para poder desarrollar el plugin correctamente este deberá estar alojado dentro de la carpeta mod de Moodle dentro de una carpeta con el nombre del plugin. Moodle proporciona en su página un template con los archivos necesarios para comenzar a desarrollar el módulo.

-Los accesos a la base de datos son especialmente críticos, para ello es necesario hacer uso de las propias funciones que provee las librerías de Moodle. Al usarlas no es necesario establecer las conexiones ni cerrarlas, si no que se encargan de ello. Tiene posibilidad de emplear sentencias de MySQL almacenadas en un string para facilitar el acceso en caso de consultas que requieran cierta complejidad.

-Cuando se desee incluir un mensaje dentro del módulo para mostrarlo. Habrá que incluirlo de la forma correcta. Para ello este debe ser incluido en tantos idiomas como se desee que disponga el módulo. Dentro de la carpeta Lang del módulo deberemos crear tantas carpetas como idiomas deseemos. Deben estar nombradas con las iniciales del idioma, por ejemplo, en ingles la carpeta se llamará EN y en español ES. Que son los dos idiomas que dispone el módulo. Dentro habrá un archivo llamado igual que el nombre que le hayamos puesto al plugin. En este caso dbquiz.php. Dentro de este es donde deberemos incluir cualquier texto que deseemos mostrar.

-El modelo de programación que se sigue es MVC, modelo vista-controlador. Así pues, dispondremos de archivos que tendrá la vista del módulo, donde se incluirán, entre otras cosas, todos los formularios necesarios para llevar a cabo el ejercicio. Por detrás el modo controlador será el encargado de tratar toda la información que se va generando.

4.3. Diseño de la base de datos

Para que el módulo disponga de todas las funcionalidades requeridas es necesario crear una base de datos con varias tablas en las que, por un lado, almacene todos los datos referentes a la creación del ejercicio, tablas, campos, preguntas, etc. Por otro lado, debe ser capaz de almacenar de forma segura todas las respuestas propuestas por los alumnos.

En el siguiente esquema veremos cómo quedaría la base de datos diseñada:

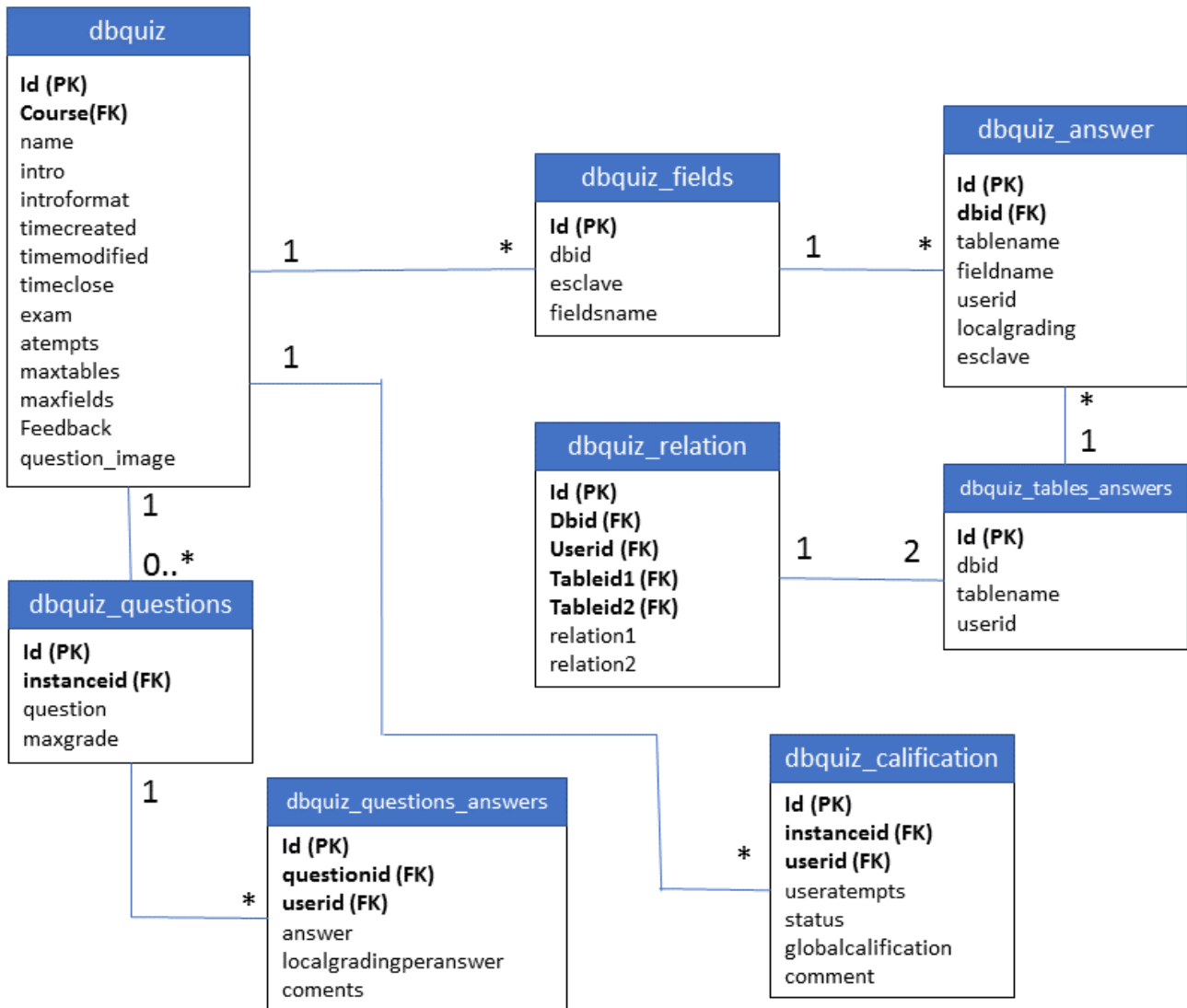


Ilustración 7: Esquema de la base de datos de Dbquiz

Carpeta db/ install.xml:

Dentro de esta carpeta encontraremos los ficheros necesarios para instalar la base de datos del módulo. Para generar la base de datos correspondiente al módulo es necesario utilizar el fichero incluido en ella llamado install.xml. Este archivo almacena todos los campos relativos a la nueva base de datos y además proporciona más información de estos, como el tipo de datos que se van a almacenar o si se trata de una clave entre otros.

Las tablas empleadas en la base de datos se detallan a continuación.

Tabla dbquiz:

Esta tabla almacena todos los datos relacionados con las instancias del módulo dbquiz. En esta tabla se almacenarán todos los campos introducidos por el profesor a la hora de crear el módulo. Es la encargada de almacenar datos como el título, descripción del ejercicio o fecha de creación.

A continuación, se van a explicar los campos que componen la tabla:

Dbquiz		
Nombre	Tipo	Descripción
ID	Int	Identificador de la instancia del módulo
Course	Int	Identificador del curso al que pertenece la instancia del módulo
Name	String	Nombre de la instancia del módulo
Intro	String	Descripción de la instancia del módulo
Introformat	Int	Formato que se le da a la descripción de la instancia del módulo
Timecreated	Int	Fecha en la que se ha creado la instancia del módulo
Timemodified	Int	Fecha en la que se ha modificado por última vez la instancia del módulo
Timeopen	Int	Fecha en la que alumno tiene abierto el ejercicio para realizarlo
Timeclose	Int	Fecha límite para que el alumno pueda realizar el ejercicio
Grade	Int	Puntuación máxima que puede obtener cada alumno en cada ejercicio propuesto
Exam	Int	Indica si el ejercicio propuesto es un examen o no
Atempts	Int	Número de intentos de los que dispone el alumno para realizar el ejercicio
Maxtables	Int	Máximo número de tablas que el alumno puede introducir para crear su base de datos
Maxfields	Int	Máximo número de campos que el alumno puede introducir en cada tabla para crear su base de datos. Se

		trata de una limitación máxima para todas la tablas con el fin de ayudar al alumno.
Feedback	String	Ayuda de la que dispone el alumno por parte del profesor a la hora de realizar el ejercicio
Question_image	String	Nombre de la imagen de la base de datos que propone el profesor para realizar consultas

Tabla 1: Tabla Dbquiz

Tabla dbquiz_fields:

En esta tabla se van a almacenar los nombres de los campos introducidos por el profesor en el script al crear la base de datos para el ejercicio. Esta tabla incluirá, a parte del nombre del campo, el identificador de la instancia del módulo al que pertenece y si se trata de una clave o no.

Los campos que conforman esta tabla son los siguientes:

dbquiz_fields		
Nombre	Tipo	Descripción
ID	Int	Identificador del campo de la base de datos
dbid	Int	Identificador de la instancia del módulo a la que pertenece el campo
esclave	Int	Indicador de si el campo es una clave
fieldsname	String	Nombre del campo de la base de datos

Tabla 2: Tabla dbquiz_fields

Tabla dbquiz_tables_answers:

En esta tabla se almacenan todos los nombres de las tablas propuestas por el alumno para realizar su ejercicio. En la tabla se almacena el nombre de la tabla propuesto por el alumno, la id de la instancia del módulo al que pertenecen y el id del alumno que incluye el nombre de la tabla.

Los campos que conforman esta tabla son los siguientes:

dbquiz_tables_answers		
Nombre	Tipo	Descripción
ID	Int	Identificador de la tabla propuesta por el alumno

		para la resolución de la base de datos
dbid	Int	Identificador de la instancia del módulo a la que pertenece la tabla
tablename	String	Nombre de la tabla propuesta por el alumno para la resolución de la base de datos
userid	Int	Identificador del alumno

Tabla 3: Tabla dbquiz_tables_answers

Tabla dbquiz_relation:

En esta tabla se almacenan las relaciones entre tablas propuestas por el alumno para la realización del ejercicio. En este campo se almacena el id de cada tabla que estable la relación y el tipo de relación que se establece en ambos extremos.

Los campos que conforman la tabla son los siguientes:

dbquiz_relation		
Nombre	Tipo	Descripción
ID	Int	Identificador de la relación entre tablas propuesta por el alumno para la resolución de la base de datos
dbid	Int	Identificador de la instancia del módulo a la que pertenece la relación entre tablas
userid	Int	Identificador del alumno
Tableid1	Int	Identificador de la primera tabla sobre la que se establece la relación
Tableid2	Int	Identificador de la segunda tabla sobre la que se establece la relación
Relation1	String	Tipo de relación que se establece en lado de la primera tabla
Relation2	String	Tipo de relación que se establece en lado de la segunda tabla

Tabla 4: Tabla dbquiz_relation

Tabla dbquiz_answers:

En esta tabla se incluye la relación que establece el alumno entre los campos y las tablas creadas. Se relacionan los campos con los nombres de las tablas que los contienen. Dentro de la tabla se

almacena, el nombre del campo y de la tabla, el id del usuario, si el campo que almacenado es una clave dentro de la tabla.

Los campos que forman la tabla son los siguientes:

dbquiz_answers		
Nombre	Tipo	Descripción
ID	Int	Identificador de respuesta propuesta por el alumno para la resolución de la base de datos
dbid	Int	Identificador de la instancia del módulo a la que pertenece la propuesta
tablename	String	Nombre de la tabla proporcionada por el alumno que contiene el campo indicado
fieldname	String	Nombre del campo que se incluye en la tabla
userid	Int	Identificador del alumno
esclave	Int	Indicador de si el campo incluido es una clave dentro de la tabla

Tabla 5: Tabla dbquiz_answers

Tabla dbquiz_calification:

En esta tabla se almacena la puntuación dada por el profesor al alumno y los comentarios dado por este al ejercicio. Se almacena también el número de intentos que lleva usados el alumno y el status del ejercicio del alumno.

Los campos que conforman la tabla son los siguientes:

dbquiz_calification		
Nombre	Tipo	Descripción
ID	Int	Identificador de la calificación dada al ejercicio
instanceid	Int	Identificador de la instancia del módulo a la que pertenece la calificación
userid	Int	Identificador del alumno
usedattempts	Int	Número de intentos empleados por el alumno en el ejercicio
status	Int	Indicador de en qué situación se encuentra el ejercicio del alumno

globalcalification	Int	Puntuación otorgada por el profesor al alumno en el ejercicio
Comment	String	Comentario aportado por el profesor a cerca de la resolución propuesta por el alumno

Tabla 6: Tabla dbquiz_calification

Tabla dbquiz_questions:

En esta tabla se almacenan las preguntas propuestas por el profesor para la resolución de la segunda parte del ejercicio. La cual consiste en realizar consultas que cumplan con las necesidades incluidas en las preguntas. En esta tabla se almacena la pregunta y la máxima puntuación de esta.

La tabla contiene los siguientes campos:

dbquiz_question		
Nombre	Tipo	Descripción
ID	Int	Identificador de pregunta propuesta por el profesor
instanceid	Int	Identificador de la instancia del módulo a la que pertenece la pregunta
question	String	Pregunta propuesta por el profesor
maxgrade	Int	Máxima puntuación que puede darse a la respuesta del alumno

Tabla 7: Tabla dbquiz_question

Tabla dbquiz_questions_answers:

En esta tabla se almacena la respuesta dada por el alumno a la pregunta propuesta por el profesor. También se almacena la puntuación obtenida por el alumno en esta respuesta y los comentarios oportunos proporcionados por el profesor.

La tabla contiene los siguientes campos:

dbquiz_questions_answers		
Nombre	Tipo	Descripción
ID	Int	Identificador de respuesta dada por el alumno
questionid	Int	Identificador de la pregunta a la que responde el alumno
userid	Int	Identificador del alumno
answer	String	Respuesta proporcionada por el alumno

Localgradingperanswer	Int	Calificación otorga a la respuesta dada por el alumno
Coments	String	Comentario dados por el profesor a cerca de la respuesta proporcionada por el alumno

Tabla 8: Tabla dbquiz_questions_answers

4.4. Archivos del módulo

En esta sección vamos a hacer una revisión de los ficheros más relevantes dentro del módulo, para ello vamos a dividir la sección en dos partes. En primer lugar, los ficheros requeridos por Moodle para el correcto funcionamiento del módulo y para terminar los archivos generados a posteriori, que completan la funcionalidad del módulo.

4.4.1. Archivos requeridos

Desde la plataforma de Moodle para que no se empiece de cero y con el fin de facilitar el trabajo a desarrolladores noveles, proveen una solución en la cual se dan unas pautas iniciales con las que empezar a desarrollar. El paquete incluye ficheros básicos para el desarrollo del módulo. A continuación, vamos a explicar cada uno de los ficheros básicos.

Fichero mod_form.php:

Antes de comenzar explicando en que consiste este fichero es necesario explicar la forma que tiene Moodle de crear y gestionar formularios. Por un lado, se crea una clase que es la que alberga todos los campos que deseemos para el formulario. Estos campos están definidos por Moodle en sus librerías y, por tanto, es necesario añadirlos conforme a esta definición y como un formulario clásico de php.

Este fichero consiste en una clase de formulario que es la que incluye todos los campos mostrados en la creación de la instancia del módulo. Inicialmente, vienen campos tipo como la descripción de la instancia o el título de la misma. Pero se puede personalizar a las necesidades de cada uno. Todos los campos que se incluyan aparecerán en el formulario de creación de la instancia y deben de estar relacionados con la tabla de la base de datos en la que se almacenan los datos de la instancia.

Subdirectorio db/:

Como explicamos anteriormente la forma de crear la base de datos del módulo reside en un archivo xml que se encuentra dentro de este directorio. Este archivo ya viene creado y trae los campos iniciales, pero para poder desarrollar nuestro módulo es necesario editarlo y ampliarlo.

Subdirectorio Lang:

Se trata del subdirectorio en el que se configuran las opciones de idioma, dentro de este deberemos proveer tantas carpetas como idiomas deseemos tener. Inicialmente solo viene el inglés. Dentro del directorio aparece una carpeta, con la abreviación internacional del idioma, EN, y dentro de esta hay un archivo llamado igual que el módulo. Este tiene configurado todo el diccionario del módulo.

Cada idioma que queramos introducir debe seguir esta estructura.

Archivo index.php:

Este archivo está configurado para que haga un indexado de todas las instancias generadas hasta el momento y que se encuentran activas. El archivo index.php indexa solo las instancias generadas en un curso. Desde el listado podemos acceder a cada una de estas instancias.

dbquizzes

Semana	dbquizzes
25 de febrero - 3 de marzo	test

Ilustración 8: Index del módulo dbquiz

Archivo view.php:

Como se comentó anteriormente, se ha desarrollado este módulo basado en MVC, la vista inicial del módulo es este archivo desde el cual cuelga toda la actividad y se distribuye entre los distintos actores relacionados con el módulo.

Como bien se ha explicado en otras secciones, Moodle divide los privilegios de los usuarios en roles. En función del rol del usuario en cuestión podremos mostrarle ciertas cosas.

A este fichero se accede inmediatamente después de haber clicado en el enlace de cualquier instancia.

Si se trata de un profesor con privilegios de administrador podrá acceder a la edición, visualización y evaluación del ejercicio. Pero nunca podrá realizar el ejercicio.

El profesor sin privilegios, es decir, que no puede editar el curso, solamente podrá ver el ejercicio creado por otro profesor y evaluar a los alumnos que hayan terminado su ejercicio.

El rol de alumno podrá ver la presentación del ejercicio con toda la información relevante. Podrá iniciar la creación de su base de datos y posteriormente la resolución preguntas. Posteriormente, cuando le otorguen una puntuación podrá visualizarla, así como los comentarios realizados por el profesor.

A continuación, vamos a ver un diagrama de flujo que aclarará más si cabe el funcionamiento de este archivo

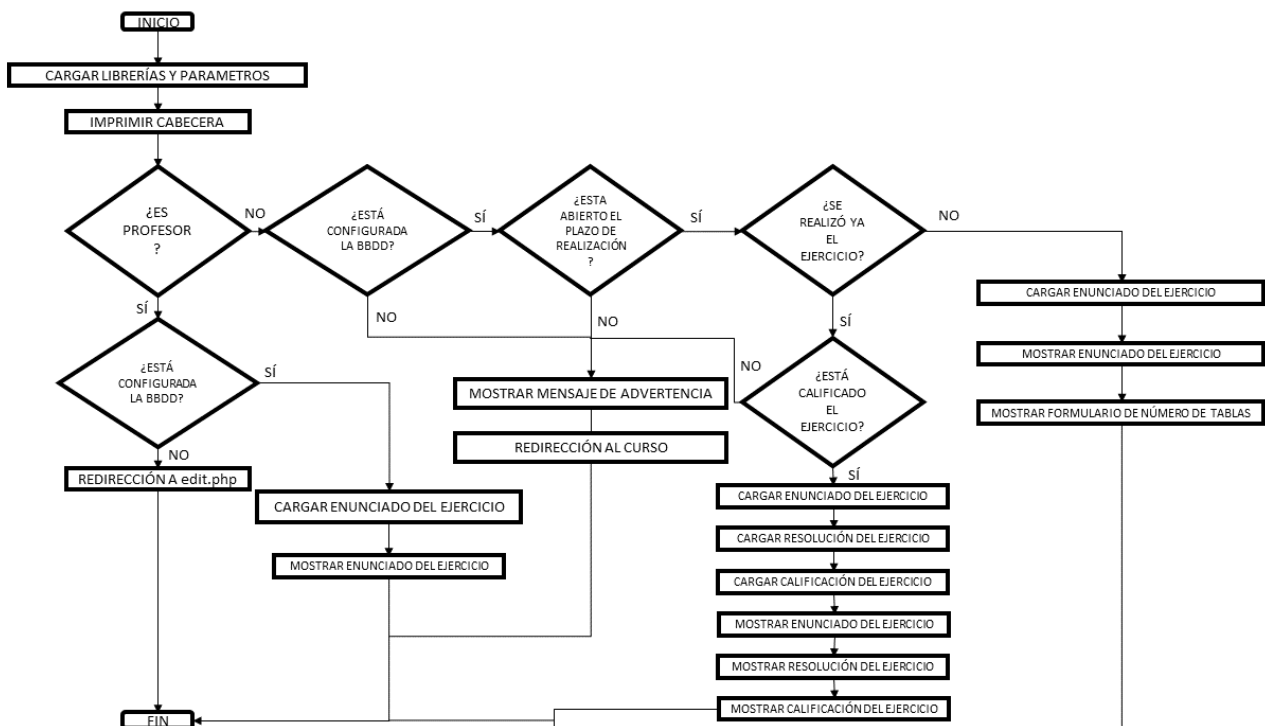


Ilustración 9: Diagrama de flujo de view.php

Archivo lib.php:

Este archivo engloba todas las funciones básicas necesarias para el correcto funcionamiento del módulo. Aunque este archivo puede ser ampliado y usado en modo de librería customizada al uso para el módulo, no es recomendable ya que existe otro fichero conocido como locallib.php, el cual veremos posteriormente, en el que el desarrollador deberá incluir sus propias funciones.

En este archivo desde el principio aparecen algunas funciones que son primordiales a la hora de constituir el módulo y que son empleadas en multitud de situaciones. Se trata de las siguientes:

dbquiz_add_instance():

Esta función es la encargada de que, por defecto, todos los campos incluido en el formulario se guarden. Es, por así decirlo, la encargada de generar una nueva instancia del módulo. También es la encargada de ligar la nueva instancia al calendario del usuario para que este permanezca actualizado.

dbquiz_update_instance():

Esta función entra en juego en el momento en el que el profesor desea hacer algún cambio en una instancia ya creada. Se encarga de recoger los cambios introducidos en la instancia y actualizar los datos en la base de datos.

dbquiz_refresh_events():

Esta función se encarga de revisar todas las instancias del módulo y de comprobar que todos los eventos generados para estas están actualizados.

dbquiz_delete_instance():

Esta función entrará en acción cuando el profesor desee borrar una instancia. En ese momento, esta función eliminará los registros de la base de datos y los eventos generados para esta instancia.

`dbquiz_user_outline()`:

Esta función es la encargada de, cuando se requiera, devuelve una instancia del módulo con la información de los pasos dados hasta el momento por un usuario en ella.

Archivo `locallib.php`:

Este archivo es el encargado de centralizar toda la actividad de controlar las funciones del módulo. En él, el desarrollador debe incluir todas las funciones creadas por él que aplican al módulo.

Este archivo está incluido en la cabecera del resto de archivos para tener disponible la librería de funciones.

Aunque este archivo esta creado por y modificado por nosotros, es un archivo básico para el módulo y viene definido como tal.

A continuación, vamos a hacer una revisión de las funciones más importantes. Creadas para el control de las acciones del módulo.

`save_fields()`:

Esta función es la encargada de almacenar los campos introducidos por el profesor en el script.

`save_questions()`:

Esta función almacena las preguntas junto con la máxima puntuación en la base de datos.

`delete_db()`:

Esta función elimina la base de datos creada a través del script introducido por el profesor. Esta función es usada cuando en el proceso de introducción del script, algo falla. Entonces se activa un mecanismo de alerta para avisar al profesor y se eliminan todas las posibles trazas que hayan quedado.

`update_esclave()`:

Esta función se encarga de añadir a la tabla de almacenaje de respuestas del alumno de creación de la base de datos que campo es una clave. Es decir, una vez que el alumno selecciona cuál es su clave primaria dentro de cualquier tabla creada por él. Esta tabla almacena dicha información.

`saving_relations()`:

Cuando el alumno establece las relaciones entre las tablas que él cree conveniente, esta función almacena el nombre de las tablas y la relación existente entre ambas.

`save_answers()`:

Esta función es la encargada de almacenar las respuestas dadas por el alumno a las preguntas propuestas por el profesor en cuanto a las consultas a la base de datos.

`reset_exercise():`

Existe la posibilidad que, llegados al punto final del ejercicio, el alumno desee volver a probar a realizar el ejercicio. Al activar esta opción se producirá un reseteo de todos los campos almacenados hasta el momento. Esta función será la encargada de realizar este borrado.

`update_userattempts():`

Cuando el alumno decida reiniciar el ejercicio para volver a completarlo. Siempre que este dispone realizar varios intentos, esta función será la encargada de actualizar el número de intentos empleados por el alumno.

`update_status():`

En función de las fases que vaya realizando el alumno el status del ejercicio ira cambiando, esta función es la encargada de realizar esta actualización.

Por otro lado, vamos a definir las fases del ejercicio.

Fase 0: El alumno no ha comenzado a realizar el ejercicio. En el momento que inicie el curso, el alumno verá el enunciado de este y todos los datos referentes a él.

Fase 1: El alumno ha realizado el 100% del ejercicio. En el momento que intente acceder al curso, se le mostrará un mensaje de que no puede volver a realizar el ejercicio porque ya lo guardó y que está a la espera de corrección.

Fase 2: El profesor ha corregido el ejercicio y lo ha puntuado. El alumno que intente acceder podrá ver la resolución de su ejercicio, con su puntuación y los comentarios añadidos por el profesor.

`update_grades_db():`

Cuando el profesor realiza la corrección de la resolución de la base de datos, esta función es la encargada de incluir la puntuación y el comentario realizado por el profesor en la tabla de calificaciones.

`update_grades_answer():`

Cada vez que el profesor otorga una puntuación al alumno en una de las preguntas respondidas. Esta función almacena esta puntuación y el comentario añadido en la tabla en la que se almacena la respuesta del alumno.

Archivo `version.php`:

Este archivo es el encargado de almacenar la versión del módulo. Moodle tiene rutinas de comprobación de actualizaciones, ya sea en modo online o en local, si este archivo posee un número de versión mayor que el que tiene el sitio almacenado, el sitio le solicitará un update al administrador del sitio para instalar la nueva actualización. Esto es interesante a la hora de desarrollar, ya que simplemente modificando este archivo podremos incluir los cambios realizados en el módulo de forma rápida y sencilla. Sin necesidad de eliminar el módulo y tener volver a instalarlo.

4.4.2. Archivos de nueva creación

Para cumplir con todas las especificaciones requeridas ha sido necesario la creación de multitud de nuevos ficheros. Como hemos comentado anteriormente la forma de crear formulario es crear una clase con todos los campos necesario y posteriormente instanciarla en otro fichero de tal forma que en este último se hace la recolección de datos del formulario y se procede al control de ellos. Por así decirlo la parte de vista de modelo vista-controlador se divide en dos partes, la clase del formulario y la instancia de la misma. Por tanto, para hacer la revisión de los ficheros de nueva creación va a ser frecuente hacerlo por pares para que se comprenda mejor como está constituido el módulo.

`edit.php / add_db_form.php`:

Este es el archivo encargado de gestionar la introducción del nuevo script por parte del profesor para la creación de la base de datos. El formulario añade un cuadro de texto donde añadir el script completo. En ese momento se inicia la creación de base de datos.

Por otro lado, si se accede a la pestaña de editar la base de datos, el archivo requerirá una confirmación para el borrado de la actual base de datos. De esta forma, lo que se realizará es un reseteo de la base de datos y permitirá de nuevo volver a crearla.

A continuación, vamos a ver un diagrama de flujo para entender mejor su funcionamiento.

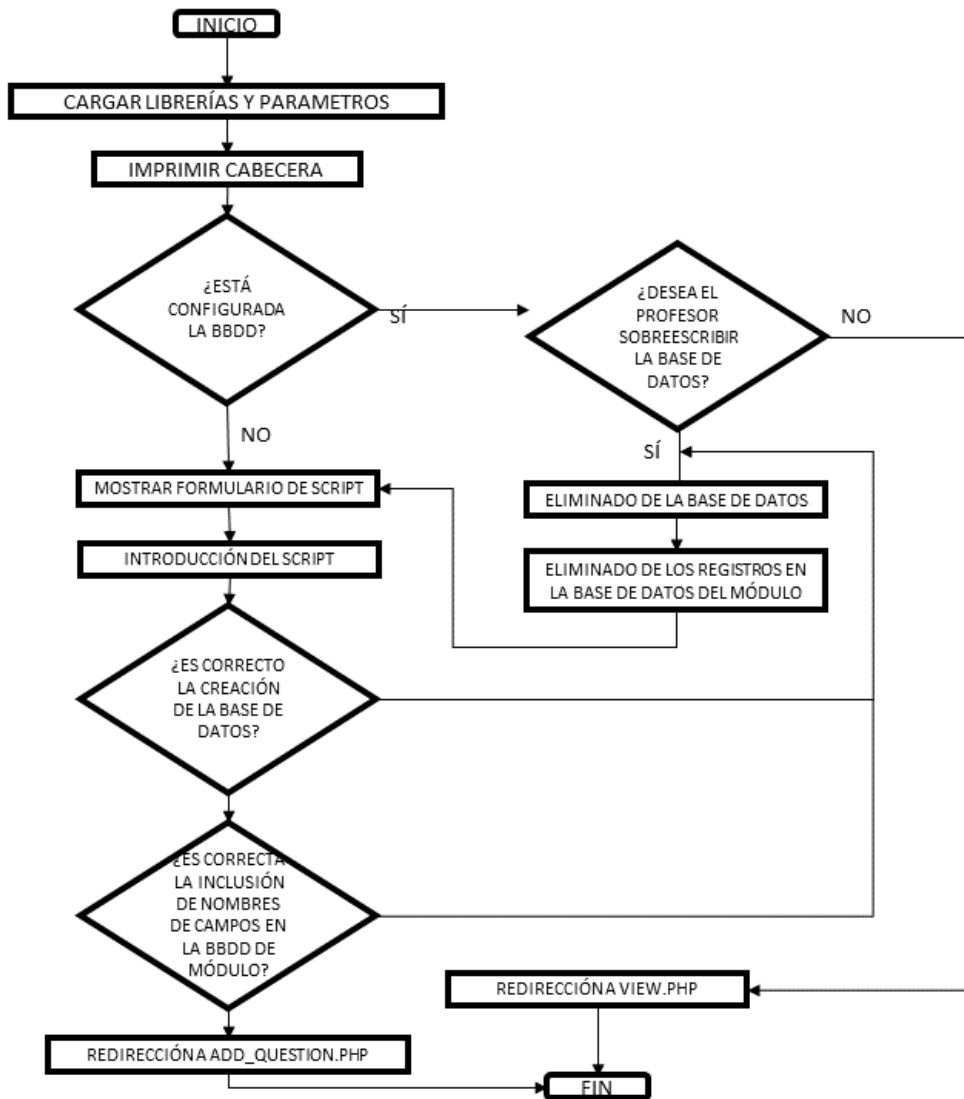


Ilustración 10: Diagrama de flujo de edit.php

add_question.php / add_question_form.php:

Este conjunto de archivos, son los encargados de proveer la funcionalidad de configuración de la segunda parte del ejercicio. La resolución de las preguntas. Por un lado, este formulario contiene un almacenador de imágenes. La cuales, una vez validadas, se almacenan en el directorio situado dentro de la carpeta del módulo llamada “images”. Por otro lado, dispone la funcionalidad de almacenar las preguntas que el profesor desee. Estas preguntas, junto con su puntuación máxima son almacenados en su tabla correspondiente.

A continuación, vamos a ver un diagrama de flujo para comprender mejor el funcionamiento de este conjunto de archivos.

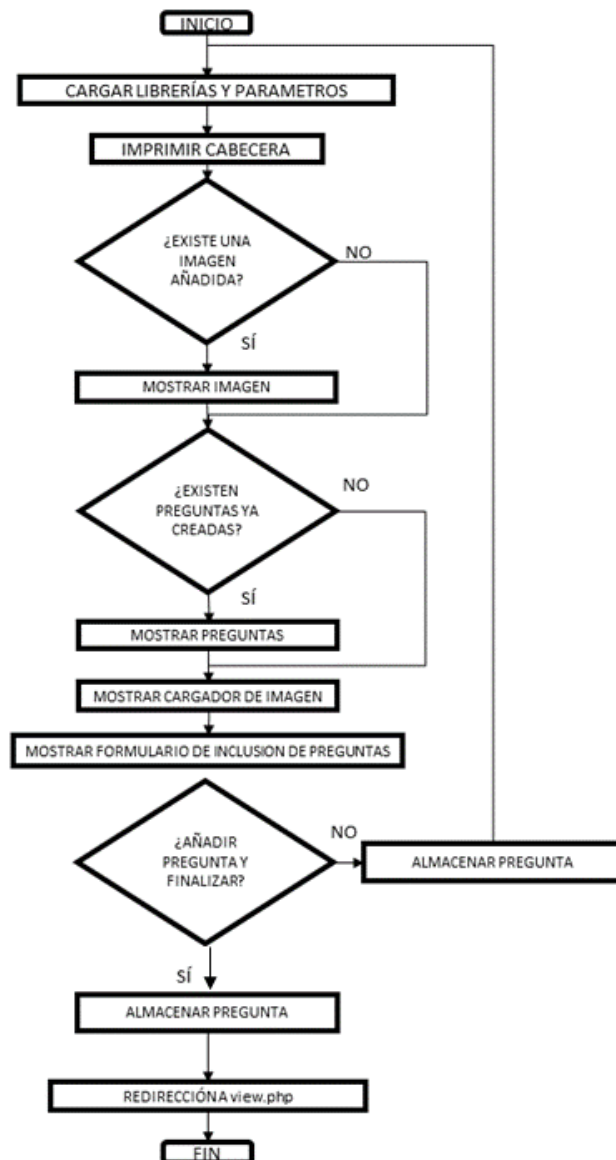


Ilustración 11: Diagrama de flujo de `add_question.php`

`Resolution.php / resolution_form.php`:

Cuando el alumno accede al ejercicio por primera vez, en el archivo `view.php` le solicita el número de tablas que desea que disponga su base de datos. Posteriormente, entra en juego el archivo `resolution.php` y el formulario. Este añade la funcionalidad de introducción de los nombres que el alumno desee a las tablas que acaba de solicitar. Estas tablas todavía no son almacenadas en la base de datos. Se traspa la información a través de la url a los siguientes pasos.

`res_tab.php / registro_campos.php / registro_tablas.php / relations_form.php`:

Este es el núcleo de la resolución del ejercicio. Se encarga de relacionar los campos con las tablas propuestas. Vamos a dividir la explicación del funcionamiento de este archivo en dos partes.

Por un lado, tenemos funcionalidad de añadir los campos deseados a las tablas anteriormente creadas. Para ello, disponemos una tabla creada con todos los campos añadidos por el profesor a través del script. Disponemos de un selector, en cual aparecen todas las tablas creadas por el

alumno. También, existe un botón para almacenar los campos. El funcionamiento es el siguiente, el alumno deberá seleccionar la tabla que quiere llenar, y después seleccionar los campos, que, a su juicio, pertenecen a dicha tabla. Posteriormente, deberá presionar el botón. Automáticamente, los campos desaparecerán de la tabla genérica y aparecerán en la tabla seleccionada. Esta funcionalidad esta creada en JavaScript, se trata de las siguientes funciones:

```
$(function(){
    $table.tablaBBDD tr").each(function(){
    $(this).click(function(){
    if($(this).attr('class') == 'fila'){
        $(this).removeClass('fila');
        $(this).addClass('seleccionado');
    }else{
        $(this).removeClass('seleccionado');
        $(this).addClass('fila');
    }
    })
    });
```

Esta función es la encargada de permitir la selección o deselección de los campos en la tabla genérica.

```
$("#pasar").click(function(){
    $("table.tablaBBDD tr").each(function(){
    if($(this).attr('class') == 'seleccionado'){
        $('# + form1.select.value).append($(this));
    }
    })
    });
```

Esta otra función, a través del DOM, se encarga de cambiar los elementos de una tabla a otra.

```
xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
} catch (E) {
    xmlhttp = false;
}
}
```



```

if (!xmlhttp && typeof XMLHttpRequest!="undefined") {
    xmlhttp = new XMLHttpRequest();
}
return xmlhttp;
}
$.ajax({
    url: './registro_tablas.php',
    type: 'POST',
    data: {dbid: cmid, tablename:namingtables".$i.", userid:usuariojs},
    success: function(data){
        console.log(data);
    },
    error: function(error){
        console.log(error);
    }
});

for(var i = 1; i <= cellsOfRow".$i.".length; i++)
{
    var aux = cellsOfRow".$i."[i];
    if(aux !== undefined){
        aux = aux.trim();
        aux = aux.toString();
        console.log(aux);
        debugger;
        $.ajax({
            url: './registro_campos.php',
            type: 'POST',
            data: {dbid: cmid, tablename:namingtables".$i.", fieldname: aux, userid:usuariojs,
localgrading: 0, esclave:0},
            success: function(data){

```

```
    console.log(data);  
  },  
  error: function(error){  
    console.log(error);  
  }  
});  
}  
}
```

Esta es la función encargada de realizar el acceso a memoria de forma asíncrona a través de AJAX. De tal manera que sin necesidad de refrescar la página tendremos almacenados los campos. Esto se realiza a través de los ficheros registro_campos.php y registro_tablas.php. Estos ficheros son los encargados de realizar las consultas a la base de datos y generar el registro. De esta forma, es cuando dejamos fijadas las respuestas del alumno.

Hemos hablado de que este fichero tenía dos funcionalidades muy importantes. La primera parte ya ha sido explicada. La segunda funcionalidad está relacionada con las relaciones que el alumno desea establecer entre tablas de la base de datos. Aquí dispondrá de un campo donde introducir el número de relaciones que va a establecer. Se trata de un simple paso de parámetros a través de URL que prepara el siguiente escalón del proceso.

A continuación, vamos a ver un diagrama de flujo para comprender mejor el funcionamiento de este complejo conjunto de archivos.

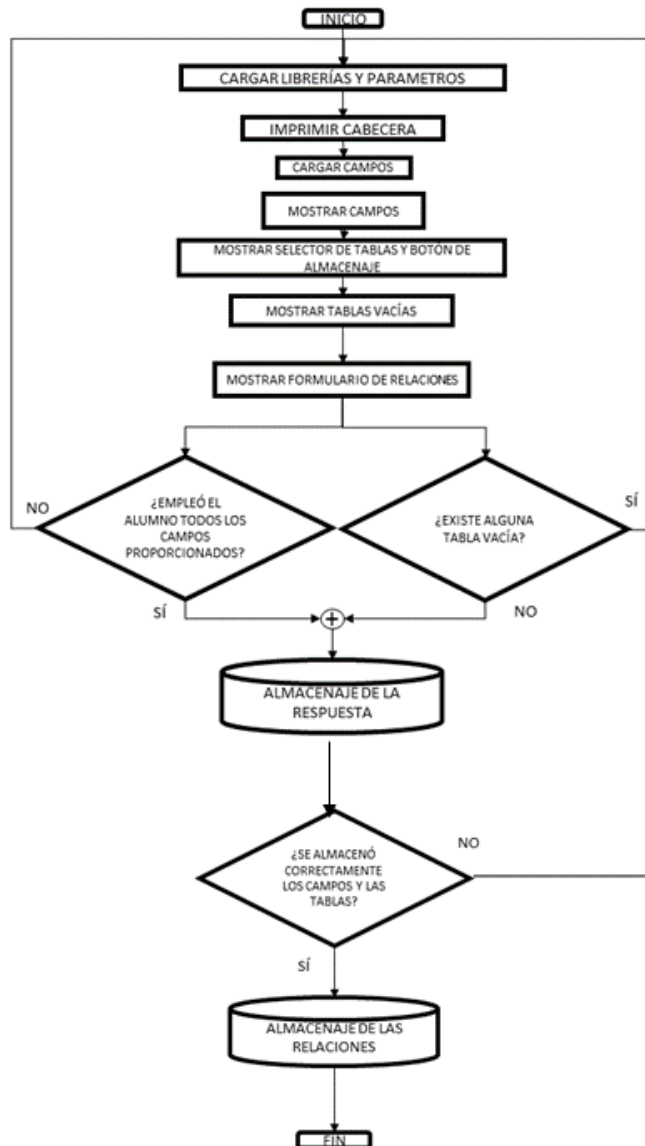


Ilustración 12: Diagrama de flujo del registro de tablas y campos

res_tab2.php / primary_key_form.php:

Este conjunto de ficheros son el siguiente escalón para permitir al alumno continuar resolviendo su ejercicio. Estos ficheros son los encargados de permitir añadir a cada tabla creada la información de cuál de sus campos es una clave primaria. También permite establecer las relaciones entre las tablas. Anteriormente, se estableció el número de relaciones, ahora creamos campos selectores donde se elige que tablas mantienen relación y que tipo.

resolution_question.php / resolution_question_form.php:

Una vez terminada la fase de creación de la base de datos, entramos en el apartado de resolución de preguntas. Si el profesor dispuso preguntas para resolver, se tendrá acceso a ellas para resolverlas, si no este paso no aparecerá. En este fichero mostramos la imagen de la base de datos proporcionada por el profesor y las preguntas. Justo debajo de cada pregunta aparece un cuadro de texto para añadir las respuestas. Una vez cumplimentados todos los campos, al validar se almacenarán estas respuestas en la tabla correspondiente.

Validation.php / validation_form.php:

Una vez realizado todo el ejercicio, llegamos a esta parte en la cual el alumno puede decidir si está conforme con su ejercicio o si por el contrario, siempre y cuando le queden intentos, volver a realizarlo. Si el alumno guarda el ejercicio y lo da por terminado. La funcionalidad de este fichero actualizará el status del ejercicio para que quede listo para ser corregido. Si por el contrario desea volver a realizarlo. Este fichero eliminará todos los datos almacenados hasta el momento relacionados con este usuario de la instancia e incrementará el contador de intentos gastados en la base de datos.

Scoring.php / scoring_form.php:

Este archivo es el encargado de comprobar que alumnos han realizado el ejercicio o no. Si lo han realizado al lado de su nombre aparecerá un radio button el cual el profesor puede seleccionar y al darle a continuar podrá evaluar a dicho alumno. Si el alumno ya ha sido evaluado al lado de su nombre aparecerá un desglose de sus calificaciones.

A continuación, vamos a ver un diagrama de flujo que explica el funcionamiento de este archivo.

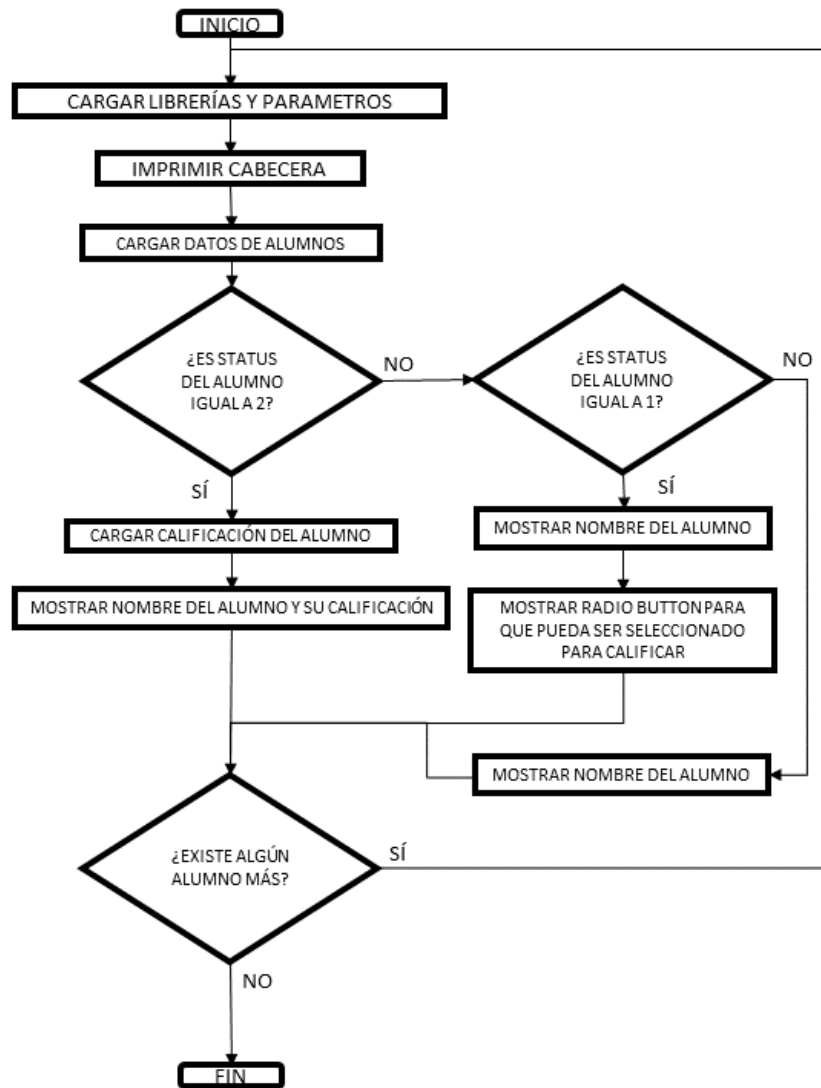


Ilustración 13: Diagrama de flujo de scoring.php

Evaluation.php / evaluation_form.php:

Estos archivos son los encargados de permitir al profesor emitir una evaluación sobre el ejercicio del alumno. El fichero, en esencia, muestra todas las respuestas del alumno. Por un lado, muestra todas las tablas con sus campos incluidos, además de la información de que campos son claves y las relaciones creadas por el alumno. Justo debajo de esta información aparecen dos cuadros de texto. Primero un pequeño para adjudicar una nota y segundo uno más grande que permite añadir un comentario aclaratorio. Cabe decir, que en el primer cuadro de texto solo se acepta un número. Si se intenta almacenar cualquier otro carácter se devolverá un error.

Por otro lado, aparecen la imagen añadida por el profesor y los enunciados de las preguntas y sus respuestas. Debajo de cada respuesta aparece, al igual que antes, dos cuadros de textos. Uno para la calificación y otro para el comentario.

El profesor una vez cumplimentada la corrección deberá pulsar el botón de continuar. Esto hará que se almacenen todos los campos del formulario en su tabla correspondiente y actualizará de nuevo el status del ejercicio. Finalmente, redireccionará al usuario a fichero scoring.php donde ya aparecerán las calificaciones correspondientes.

5. Manual de usuario

5.1. Instalación

Para la correcta instalación del módulo es necesario seguir los siguientes pasos en el servidor:

1. Copiar la carpeta dbquiz, que contiene los archivos del módulo, dentro del directorio mod dentro del servidor que contiene Moodle. La dirección sería la siguiente `./moodle/mod`.

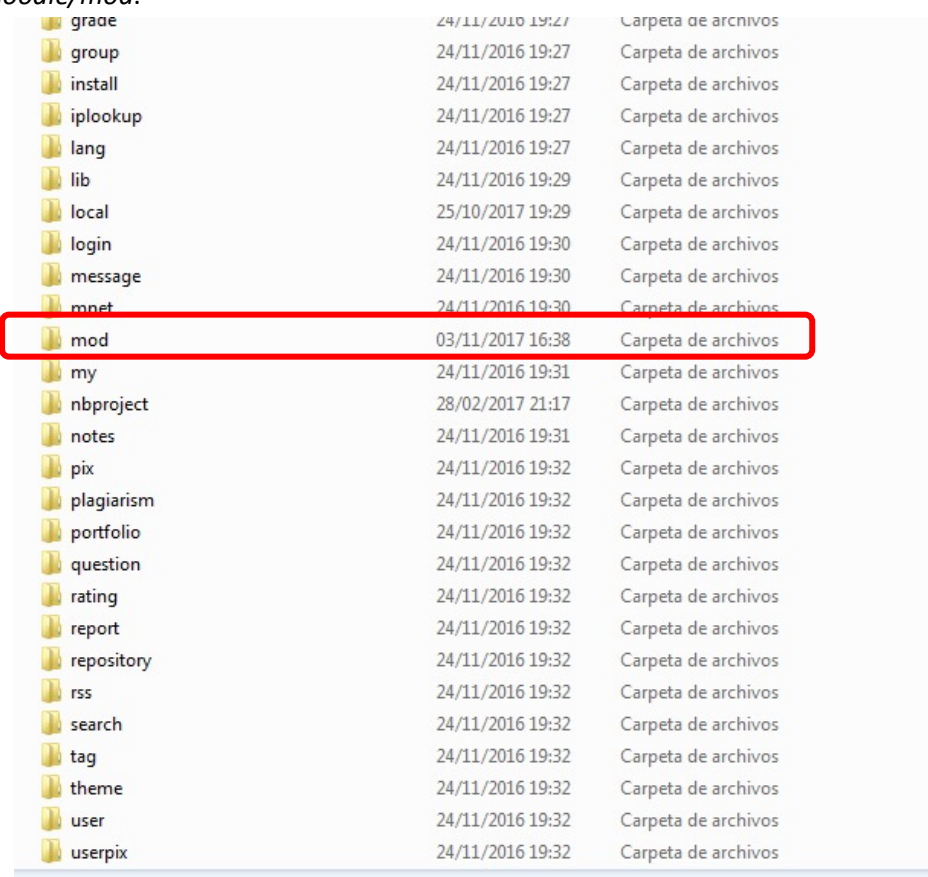


Ilustración 14: Directorio donde se aloja el módulo para ser instalado

2. Abrir moodle con el perfil de administrador. Es esperado por parte de la plataforma que, a detectar un nuevo módulo dentro de la carpeta mod, solicite la instalación del módulo.

Comprobación de 'plugins'

Esta página muestra las extensiones (plugins) que pueden requerir su atención durante la actualización, tales como nuevos plugins para ser instalados, plugins para ser actualizados, plugins ausentes, etc. Los plugins adicionales (módulos externos o add-ons) son mostrados si existe una actualización para ellos. Se recomienda que compruebe si hay disponible versiones más recientes de los módulos externos y actualice su código fuente antes de continuar con esta actualización de Moodle.

[Compruebe actualizaciones disponibles](#)

Última comprobación realizada el 2 de marzo de 2017, 13:34

Plugins solicitando atención

[Instalar actualizaciones disponibles \(1\)](#)

[Cancelar nuevas instalaciones \(1\)](#)

[Plugins solicitando atención \(2\)](#)

[Todos los plugins \(411\)](#)



Nombre de la extensión / Directorio	Versión actual	Nueva versión	Requiere	Origen / Estado
Módulos de actividad				
 Cuestionario activo <small>/mod/activequiz</small>	2016030601	2016030601	• Moodle 2015051100	Adicional Instalado ¡Existe una nueva versión 2017010400 disponible! Liberado 3.7.1 (Build: 2017010400) Versión estable Descargar Más info... Instalar esta actualización
 dbquiz <small>/mod/dbquiz</small>		2007040210	• Moodle 2014051200	Adicional Para instalarse Cancelar esta instalación

Ilustración 15: instalación del módulo

Si esto no sucediera, es necesario acceder a notificaciones. Allí aparecerá todos los módulos que necesitan ser actualizados o los que están listos para ser actualizados. Simplemente será necesario actualizar la base de datos



Ilustración 16: Pestaña lateral de administración del sitio

Si ninguno de estos métodos resulta, en la pestaña extensiones, aparecen todos los módulos incluidos. Desde ahí, se instalaría el módulo fácilmente.

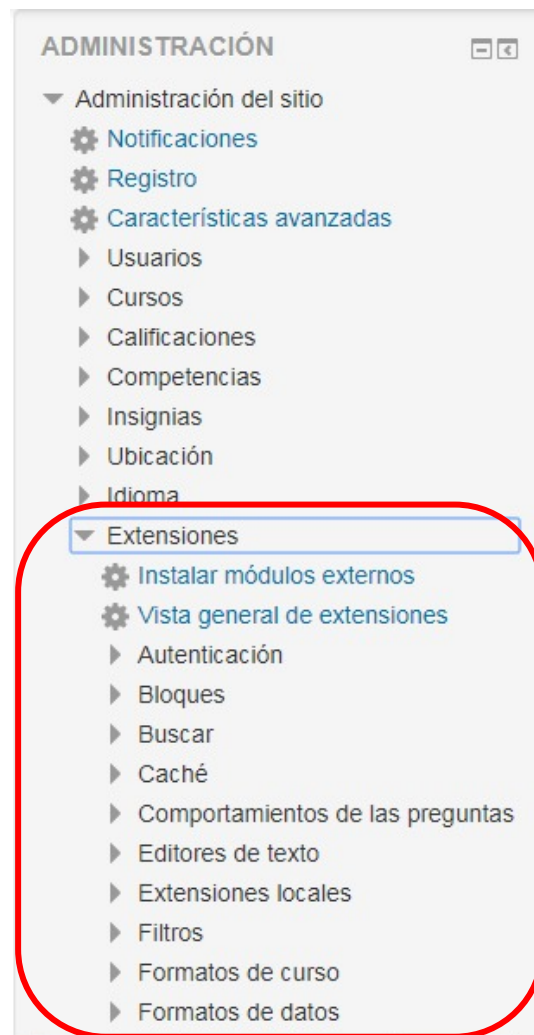


Ilustración 17: Apartado para gestionar la instalación del módulo

Una vez se haya instalado, recibiremos un mensaje de validación de la instalación.

Actualizando la versión

mod_dbquiz

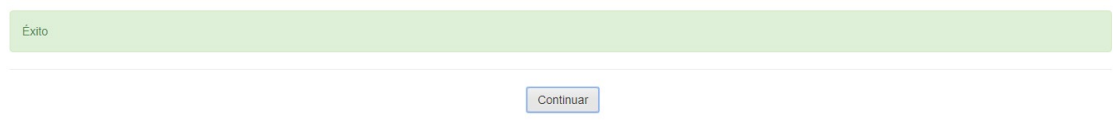


Ilustración 18: Confirmación de instalación del módulo

3. Comprobar que el módulo ha sido instalado correctamente yendo a un curso y, añadiendo una nueva actividad, ver si aparece en el listado.

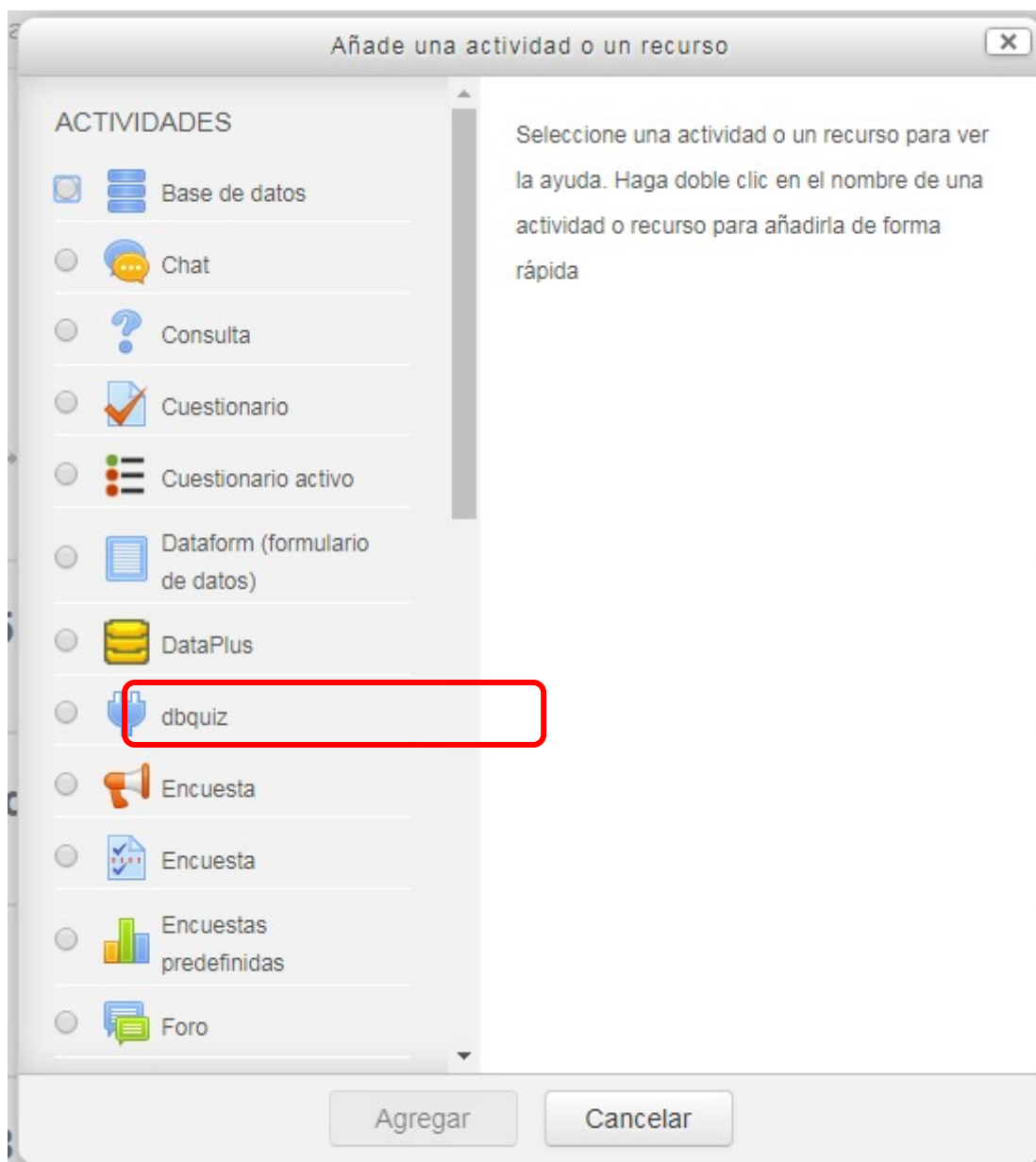


Ilustración 19: Creación de una instancia

Configuraciones para el correcto funcionamiento del módulo

1. Editar el archivo registro_campos.php. Este fichero es el encargado de almacenar los campos cuando el alumno elabora su base de datos. Esta fuera de las librerías de moodle por tanto es necesario customizarlo para adecuarlo a la base de datos que vaya a contener el módulo.
2. Editar el archivo registro_tablas.php. Este fichero almacena las tablas propuestas por el alumno para la resolución de la base de datos. Al igual que el anterior, es necesario adecuarlo a la base de datos.

```

<?php

$dbid = $_POST['dbid'];
$tablename = $_POST['tablename'];
$userid = $_POST['userid'];
$fieldname = $_POST['fieldname'];
$localgrading = $_POST['localgrading'];
$esclave = $_POST['esclave'];
$con = mysqli_connect('localhost', 'root', '', 'moodle');
if($con->connect_error)
{
    die("Connection failed:" . $con->connect_error);
}
echo($_POST['fieldname']);
$sql='insert into mdl_quiz_answers (dbid, tablename,
echo($sql);
if($con->query($sql)===TRUE)
{
    echo "OK!";
}else
{
    echo "ERROR " . $sql . "<br> " . $con->error;
}
$con->close();

?>

```

Ilustración 20: Apoyo gráfico de las recomendaciones en la instalación

Es necesario modificar lo que se encuentra dentro del recuadro de la imagen. Es necesario incluir los datos del host que albergará moodle.

5.2. Creación y configuración de una instancia

1. Acceder al sitio moodle con el perfil de administrador o en su defecto con el del profesor administrador del curso
2. Ingresar en el curso que se desea crear la instancia.
3. Dentro del curso, es posible que no esté habilitada la edición del curso. Para hacerlo, en la columna de la derecha deberemos hacer clic en el botón 'Activar Edición'.



Ilustración 21:Pestaña de activación de edición del curso

4. En el curso, aparecerá en cada semana la opción 'Añade una actividad o un recurso'. Es necesario hacer clic en ella.

The screenshot shows a course management interface with a calendar view. At the top left, there is a plus icon, a speech bubble icon, and the text "Avisos" with a pencil icon. At the top right, there are two "Editar" (Edit) buttons, one with a dropdown arrow and one with a user profile icon. Below these are three horizontal sections, each representing a date range:

- Section 1: "+ Añade una actividad o un recurso" (Add an activity or resource)
- Section 2: **25 de febrero - 3 de marzo** (with a plus icon and a pencil icon) and "Editar" (with a dropdown arrow)
- Section 3: **4 de marzo - 10 de marzo** (with a plus icon and a pencil icon) and "Editar" (with a dropdown arrow)
- Section 4: **11 de marzo - 17 de marzo** (with a plus icon and a pencil icon) and "Editar" (with a dropdown arrow)

Each date range section is followed by a horizontal line and the text "+ Añade una actividad o un recurso".

Ilustración 22: Curso preparado para crear una nueva instancia

5. Una vez realizado el anterior paso aparecerá una ventana emergente donde deberemos seleccionar entre todas las actividades y recursos el dbquiz.

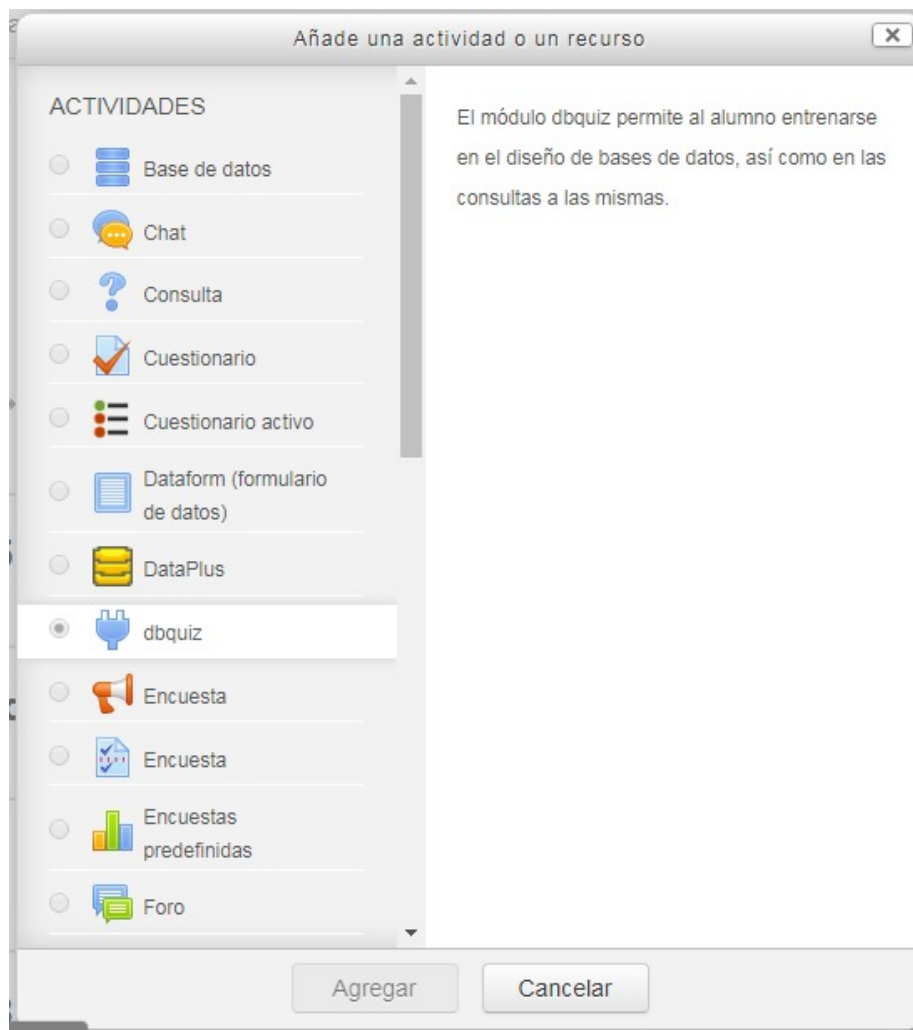


Ilustración 23: Creación de la instancia

6. A continuación, tendremos que añadir el nombre de la instancia. Este nombre será el que tomará el ejercicio.

Nombre de dbquiz*

7. Introducir la descripción del ejercicio. Esta descripción será el enunciado del ejercicio que aparecerá en la presentación al alumno.



Ilustración 24: Cuadro de texto para incluir la descripción del ejercicio

8. Introducir el Ayuda. Si el profesor considera que es necesario incluir ciertas pistas para el alumno, se pueden incluir aquí. Estas serán mostradas si el alumno decide repetir el ejercicio.

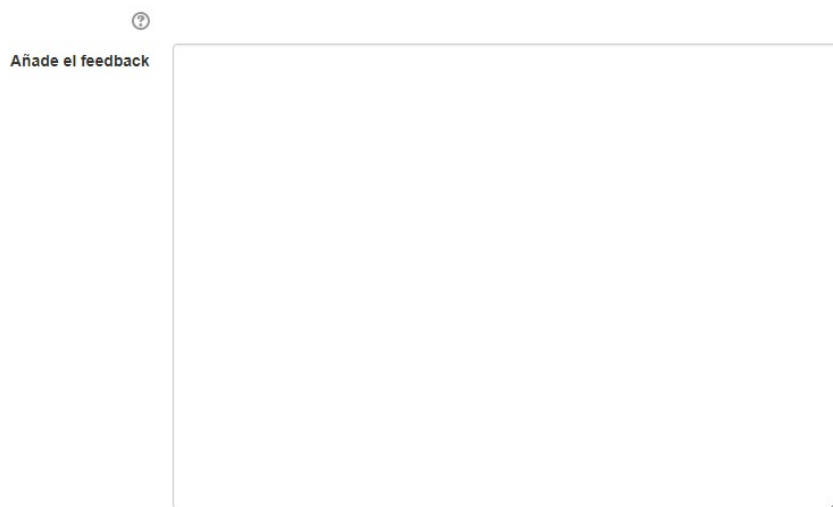


Ilustración 25: Cuadro de texto para incluir las pistas en el ejercicio

9. Ajustes de calendario. Será necesario establecer dos fechas. La primera será la apertura del ejercicio, la cual marca el momento en que se puede comenzar a realizar el ejercicio. Por otro lado, está la fecha de cierre que indica el final del tiempo de realización del ejercicio.

Apertura del ejercicio* 12 ▼ diciembre ▼ 2017 ▼ 11 ▼ 10 ▼ 

Cierre del ejercicio* 12 ▼ diciembre ▼ 2017 ▼ 11 ▼ 10 ▼ 

Ilustración 26: Selectores para añadir fechas de apertura y cierre de ejercicio

10. Introducir máximo de tablas y de campos. Estas cotas máximas serán las encargadas de apoyar al alumno a la hora de realizar su ejercicio, impidiendo que este se exceda a la hora de crear tablas o introducir demasiados campos en estas.

Número máximo de tablas*

Número máximo de campos*

Ilustración 27: Cuadros de texto donde añadir número máximo de tablas y campos

11. Ajustes de calificaciones. En este apartado se introducen tres aspectos fundamentales a la hora de configurar el modelo de calificación. Por un lado, está la opción de convertir el ejercicio en un examen. En el cual, solo se dispone de un intento y no existe la posibilidad de visualizar la ayuda. Por otro lado, se añade el máximo de puntuación que se puede otorgar y el número de intentos que el alumno puede hacer.

Examen

Puntuación máxima

Número de intentos

Ilustración 28: Cuadros de texto para añadir puntuación máxima y el número de intentos

12. Ajustes comunes del módulo, en este apartado se puede configurar la visibilidad de la instancia. Si se desea mantener oculta por el momento o dejarla visible. Se puede añadir un id propio si se desea o adjuntarla a algún determinado grupo.

Visible

Número ID

Modo de grupo

Ilustración 29: Ajustes comunes de la instancia

13. Una vez configurados todos los campos, moodle nos da la opción de guardar la configuración y permanecer en el curso o directamente acceder a la instancia para terminar de configurarla.

Ilustración 30: Botones de guardado

5.3. Creación de la base de datos

1. Acceder a la instancia creada anteriormente.



Ilustración 31: Instancia creada en el curso

2. Una vez dentro, al no estar configurada, nos redirige a la pestaña de edición.



(Continuar)

Ilustración 32: Redirección a edición de base de datos

3. En la pestaña de edición, aparecerá un cuadro de texto donde podremos introducir el script de creación de la base de datos.

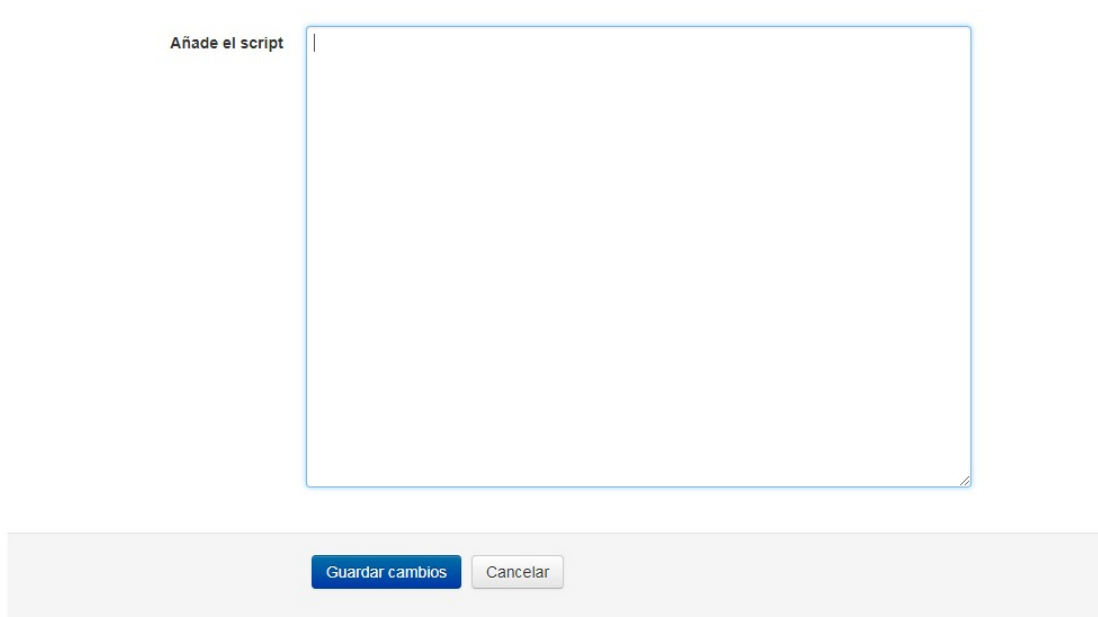


Ilustración 33: Cuadro de texto para añadir el script de creación de base de datos

4. Una vez incluido deberemos hacer clic en el botón guardar cambios.

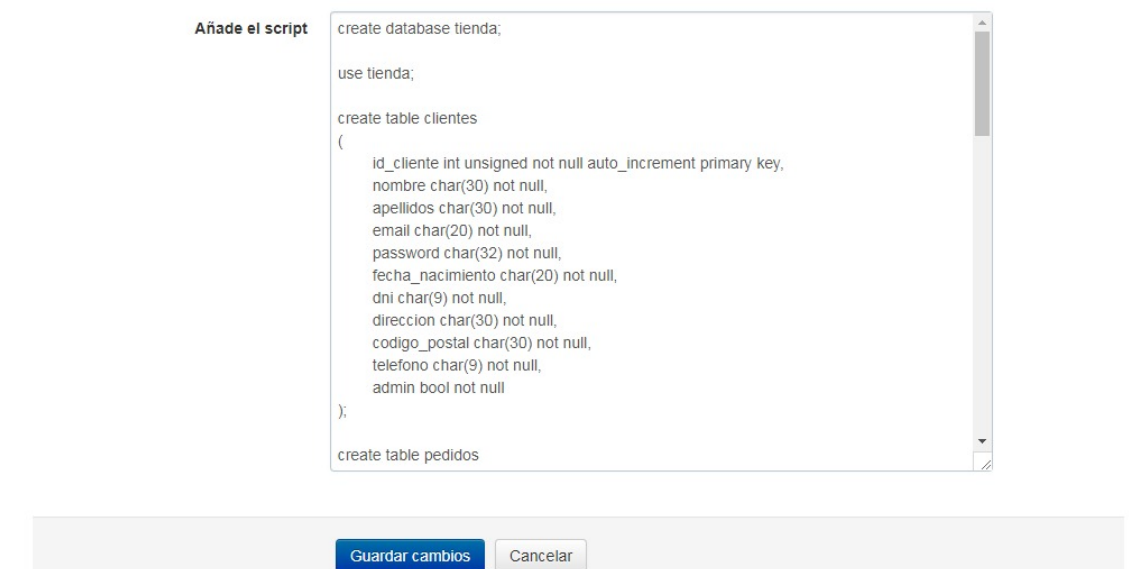


Ilustración 34: Script incluido en el cuadro de texto

5. Si ahora vamos al apartado de vista de la base de datos podremos comprobar cómo se ha creado correctamente la base de datos.

Base de datos:

Nombre: trabajo final de grado

Tablas:

1 album
2 clientes
3 pedido_album
4 pedidos

Campos:

1 id_album
2 titulo
3 grupo
4 genero
5 precio
6 fecha_album
7 caratula
8 ventas
9 oferta
10 lista_canciones
11 id_cliente
12 nombre
13 apellidos
14 email
15 password
16 fecha_nacimiento
17 dni

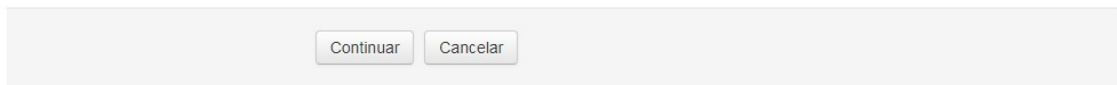
18 direccion
19 codigo_postal
20 telefono
21 admin
22 id_pedido
23 id_album
24 cantidad
25 id_pedido
26 id_cliente
27 coste_total
28 fecha

Ilustración 35: Presentación de la base de datos añadida

5.4. Edición de la base de datos

Si la base de datos ya estuviese creada, al entrar en edición nos preguntará si deseamos editar la base de datos. De tal forma que, al aceptar, se eliminará la base de datos anteriormente creada para permitir introducir de nuevo el script.

Usted va a proceder a editar la base de datos. En el momento que haga clic en continuar se sobrescribirán los datos anteriores de la base de datos y las preguntas asociadas. ¿Desea continuar?



Continuar Cancelar

Ilustración 36: Mensaje de edición de la base de datos

5.5. Creación de preguntas

1. Acceder a la pestaña de añadir preguntas. Una vez que se ha creado la base de datos se redirige automáticamente, pero si se deseara añadir nuevas preguntas simplemente sería necesario acceder a la pestaña y se podía continuar añadiendo.

- Añadir una imagen que será la que se muestre al alumno cuando vaya a contestar las preguntas.



Ilustración 37: File manager

- Añadir la pregunta que se desea introducir

▼ Introducción de la imagen de la base de datos

Añade una pregunta*

Añade la puntuación de la pregunta

Ilustración 38: Cuadro de texto para añadir las cuestiones y su puntuación

- Una vez introducido todo podemos seguir introduciendo preguntas con el botón guardar e introducir de nuevo o bien terminar y ser redirigidos a la vista de la base de datos.

Según van siendo añadidas las preguntas, estas se van visualizando junto a la foto dentro de la pestaña de añadir preguntas. Justo por encima del formulario.

Cuestiones:



Question	Grade
Obtenga el nombre de los discos que pertenecen a categoría de rock, que tengan un precio entre 10 y 20 euros	1
Obtenga el nombre de los clientes que hayan comprado el disco American Idiot y hayan nacido antes de 1991	1
ELimina el album que valga más de 30 euros y pertenezca a la categoría Indie	1

Ilustración 39: Ejemplo de creación de preguntas

Una vez terminado, vamos a poder visualizar la imagen y las preguntas junto con la base de datos creada en la página de visualización de la base de datos creada.

Ver base de datos Editar Base de datos **Añadir preguntas** Puntuación



Question	Grade
Oblenga el nombre de los discos que pertenecen a categoría de rock, que tengan un precio entre 10 y 20 euros	1
Oblenga el nombre de los clientes que hayan comprado el disco American Idiot y hayan nacido antes de 1991	1
ELimina el album que valga más de 30 euros y pertenezca a la categoría Indie	1

Ilustración 40: Visualización de las preguntas junto con la imagen

5.6. Resolución del ejercicio

1. En primer lugar, es necesario aclarar que sólo los usuarios con perfil de alumno podrán resolver el ejercicio. Por tanto, es necesario acceder a moodle con un perfil de alumno.
2. Acceder al curso que contiene la instancia creada.



Avisos

25 de febrero - 3 de marzo



trabajo final de grado

4 de marzo - 10 de marzo

11 de marzo - 17 de marzo

18 de marzo - 24 de marzo

Ilustración 41: Visualización del curso con la instancia creada con el rol de alumno

3. Acceder a la instancia creada. Dentro de ella el alumno podrá ver el enunciado del curso con todos los datos relevantes.

trabajo final de grado

Enunciado:

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

Máxima puntuación:

10

Examen:

NO

Intentos:1

Campos:

1	id_album
2	titulo
3	grupo
4	genero
5	precio
6	fecha_album
7	caratula
8	ventas
9	oferta
10	lista_canciones
11	id_cliente
12	nombre
13	apellidos
14	email
15	password
16	fecha_nacimiento
17	dni
18	direccion
19	codigo_postal
20	telefono
21	admin
22	id_pedido
23	id_album
24	cantidad
25	id_pedido
26	id_cliente
27	coste_total
28	fecha

Ilustración 42: Presentación del ejercicio al alumno

4. Introducir el número de tablas que se desean crear por parte del alumno. Es importante notar en este punto, que, si se supera el máximo de tablas que el profesor añadió como límite, el ejercicio no avanzará y aparecerá un mensaje añadiendo el máximo de tablas permitido.

Añade el número de tablas necesarias para crear tu base de datos.

Número de tablas*

Continuar

Cancelar

En este formulario hay campos obligatorios *.

Ilustración 43: Cuadro de texto para añadir el número de tablas

Mensaje de máximo de tablas:

Añade el número de tablas necesarias para crear tu base de datos.

Número de tablas*

En este formulario hay campos obligatorios *.

* Max 5 tablas.

Ilustración 44: Visualización del número máximo de tablas a incluir

- Introducir el nombre de las tablas a crear. El ejercicio avanza a la siguiente fase. En este punto aparecerán tantos cuadros de texto como número de tablas se ha introducido en la anterior fase. Cabe notar que aparece un disclaimer en el que se indica la forma correcta de añadir los nombres de las tablas.

Proyecto final de carrera

Área personal > PFC > 25 de febrero - 3 de marzo > trabajo final de grado

NAVEGACIÓN

- Área personal
 - Inicio del sitio
 - Páginas del sitio
- Curso actual
 - PFC
 - Participantes
 - Insignias
 - General
 - 25 de febrero - 3 de marzo
 - trabajo final de grado**
 - 4 de marzo - 10 de marzo
 - 11 de marzo - 17 de marzo
 - 18 de marzo - 24 de marzo
 - 25 de marzo - 31 de marzo
 - 1 de abril - 7 de abril
 - 8 de abril - 14 de abril
 - 15 de abril - 21 de abril
 - 22 de abril - 28 de abril

Añade el nombre de cada una de las tablas de tu base de datos.
IMPORTANTE: Los nombres de las tablas solo podrán tener caracteres alfanuméricos.

▼ Añadir nombres de tablas

Añade un nombre*

Añade un nombre*

Añade un nombre*

Añade un nombre*

En este formulario hay campos obligatorios *.

Ilustración 45: Cuadros de texto para darle nombre a las tablas que se van a crear

- Adjuntar los campos a las tablas creadas. En este punto, el alumno deberá incluir todos los campos en las tablas anteriormente creadas. Para ello, dispone de un listado de los campos y de un selector para elegir la tabla donde van incluidos dichos campos.

Seleccione todos los campos que desee incluir en una tabla y pulse el botón Añadir campos a la tabla. Una vez todas las tablas tengas los campos asociados. Pulse el botón de almacenar campos.

Campos:

id_cliente
id_album
coste_total
cantidad
telefono
id_album
ventas
caratula
admin
fecha_album
oferta
lista_canciones
id_pedido
precio
genero
nombre
codigo_postal
direccion
id_cliente
dni

id_pedido
fecha_nacimiento
apellidos
grupo
email
titulo
password
fecha

Seleccione ... ▼

Añadir campos a la tabla

Ilustración 46: Presentación de los campos y los selectores para incluir dichos campos en las tablas anteriormente creadas

Es importante notar que es obligatorio incluir todos los campos en las tablas, si esto no sucediese y se presionase el botón de guardar, aparecerá una alerta.

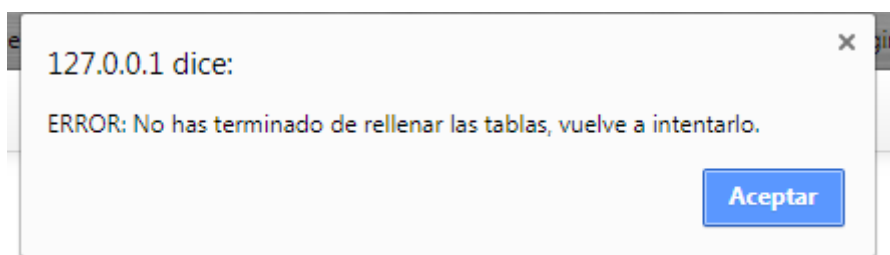


Ilustración 47: Mensaje de error indicando que quedan campos por agregar a las tablas

Del mismo modo, si alguna tabla queda sin introducir ningún campo en ella, aparecerá una alerta y volverá al punto de partida de inclusión de campos.

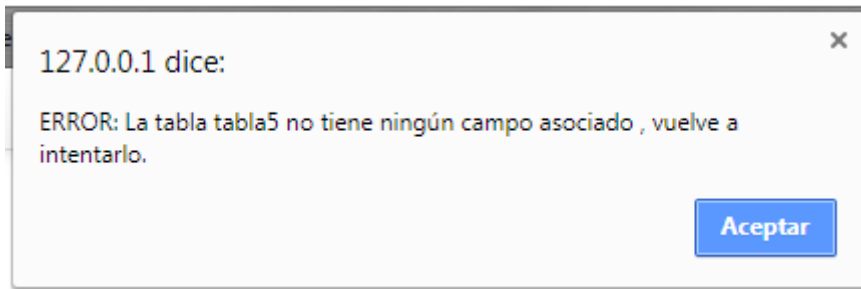


Ilustración 48: Mensaje de error indicando que una tabla ha quedado vacía

Las tablas deben tener un máximo de campos incluidos. Si se sobrepasase ese límite, también aparecerá una alerta.

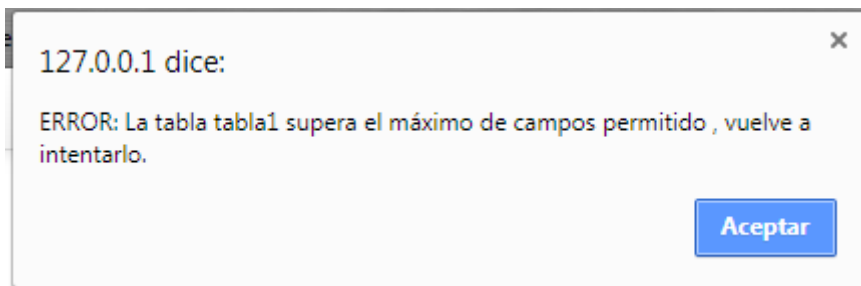


Ilustración 49: Mensaje de error indicando que se ha superado el número máximo de campos que se pueden incluir en las tablas

Con las tablas completas correctamente se podrá continuar.

cliente
password
fecha_nacimiento
telefono
nombre
dni
id_cliente
direccion
email
admin
codigo_postal
apellidos

album
ventas
id_album
genero
titulo
precio
lista_canciones
oferta
grupo
caratula
fecha_album

datospedido
id_pedido
id_album
cantidad

pedidos
id_pedido
id_cliente
coste_total
fecha

Ilustración 50: Tablas creadas con sus campos añadidos

- Una vez almacenados correctamente los campos. Volverá a aparecer las tablas vacías y ningún campo para seleccionar. Es el momento de introducir el número de relaciones que se establecen entre tablas.

Número de relaciones*

Continuar

Cancelar

En este formulario hay campos obligatorios *.

Ilustración 51: Cuadro de texto para añadir el número de relaciones entre tablas

- Introducir las claves primarias y las relaciones. En el siguiente paso, aparecerán todas las tablas creadas con sus campos y al lado de estos un check button. El alumno deberá seleccionar el check button del campo que considera que es una clave en cada tabla.

clientes

admin	<input type="checkbox"/>
fecha_nacimiento	<input type="checkbox"/>
direccion	<input type="checkbox"/>
codigo_postal	<input type="checkbox"/>
email	<input type="checkbox"/>
telefono	<input type="checkbox"/>
id_cliente	<input checked="" type="checkbox"/>
apellidos	<input type="checkbox"/>
nombre	<input type="checkbox"/>
password	<input type="checkbox"/>
dni	<input type="checkbox"/>

Ilustración 52: Selección de las claves dentro de una tabla

9. Introducir las relaciones entre tablas. Debajo de la selección de claves, aparecerán un conjunto de selectores en los cuales aparecerán los nombres de las tablas y las posibles relaciones a establecer en ambos extremos.

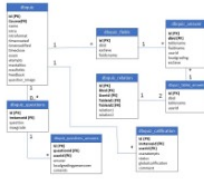
Introduce las relaciones entre tablas

clientes	▼	1	▼	1..*	▼	pedidos	▼
album	▼	1..*	▼	1	▼	pedidos	▼
pedidos	▼	1	▼	1	▼	datospedido	▼

Continuar Cancelar

Ilustración 53: Selectores para establecer relaciones entre tablas

10. En este punto, entramos en la segunda fase del ejercicio. La resolución de preguntas. Si el profesor no hubiese introducido preguntas, aquí acabaría el ejercicio. Pero si están incluidas, aparecerá una nueva página encabezada por la imagen introducida por el profesor. Debajo, aparecerán las preguntas y debajo de cada pregunta un texto para introducir la respuesta.



Responde a la siguientes preguntas

Obtenga el nombre de los discos que pertenecen a categoría de rock, que tengan un precio entre 10 y 20 euros

Respuesta: *

Obtenga el nombre de los clientes que hayan comprado el disco American Idiot y hayan nacido antes de 1991

Respuesta: *

Elimina el album que valga más de 30 euros y pertenezca a la categoría Indie

Respuesta: *

Continuar

Cancelar

Ilustración 54: Cuadros de texto para añadir las respuestas a las preguntas planteadas

- Una vez concluido el ejercicio, aparecerá todo el ejercicio creado. Aquí el alumno deberá decidir si está conforme con su ejercicio o prefiere volver a realizarlo.

clientes

Nombre del campo	Primary key
admin	
fecha_nacimiento	
direccion	
codigo_postal	
email	
telefono	
id_cliente	Sí
apellidos	
nombre	
password	
dni	

album

Nombre del campo	Primary key
genero	
oferta	
id_album	Sí
caratula	
grupo	
ventas	
fecha_album	
precio	
lista_canciones	
titulo	

pedidos

Nombre del campo	Primary key
id_pedido	Sí
id_cliente	
coste_total	
fecha	

datospedido

Nombre del campo	Primary key
id_pedido	Sí
id_album	
cantidad	

Relaciones

Nombre de la tabla 1	Tipo de relación tabla 1 -> tabla 2	Tipo de relación tabla 2 -> tabla 1	Nombre de la tabla 2
clientes	1	1..*	pedidos
album	1..*	1	pedidos
pedidos	1	1	datospedido

Resolución de preguntas



Obtenga el nombre de los discos que pertenecen a categoría de rock, que tengan un precio entre 10 y 20 euros

```
select titulo from album where genero='rock' and precio between(10 and 20);
```

Obtenga el nombre de los clientes que hayan comprado el disco American Idiot y hayan nacido antes de 1991

```
select nombre from clientes where id_cliente IN(select id_cliente from pedidos where id_pedido IN(select id_pedido from pedido_album where id_album IN(select id_album from album where titulo='American idiot')));
```

ELimina el album que valga más de 30 euros y pertenezca a la categoría Indie

```
delete from album where precio > 30 and genero='Indie';
```

Ilustración 55: Validación de la resolución del ejercicio

Si dispone de varios intentos y no los ha consumido aparecerá un botón de reset.

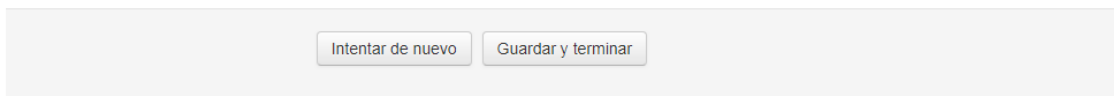


Ilustración 56: Botones de validación

Si no, simplemente aparecerá un botón de guardar y terminar.

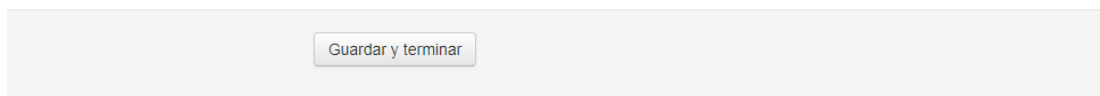


Ilustración 57: Botones de validación

12. Si se desea volver a realizar, aparecerá de nuevo la página principal, pero con el feedback mostrado. Si, por el contrario, se valida el ejercicio, quedará guardado y se redirigirá a la página del curso.
13. Si deseamos volver a entrar en el ejercicio aparecerá el mensaje de que el ejercicio ya ha sido realizado y que está a la espera de calificación y se le redirigirá al curso.

Proyecto final de carrera

Ilustración 58: Presentación del ejercicio una vez realizado y a la espera de corrección

14. Una vez que esta corregido el alumno podrá ver sus calificaciones.

Resolución de base de datos

clientes

Nombre del campo	Primary key
admin	
fecha_nacimiento	
direccion	
codigo_postal	
email	
telefono	
id_cliente	Sí
apellidos	
nombre	
password	
dni	

album

Nombre del campo	Primary key
genero	
oferta	
id_album	Sí
caratula	
grupo	
ventas	
fecha_album	
precio	
lista_canciones	
titulo	

pedidos

Nombre del campo	Primary key
id_pedido	Sí
id_cliente	
coste_total	
fecha	

datospedido

Nombre del campo	Primary key
id_pedido	Sí
id_album	
cantidad	

Relaciones

Nombre de la tabla 1	Tipo de relación tabla 1 -> tabla 2	Tipo de relación tabla 2 -> tabla 1	Nombre de la tabla 2
clientes	1	1..*	pedidos
album	1..*	1	pedidos
pedidos	1	1	datospedido

Puntuación: 7

Comentario: Perfecto!

Ilustración 59: Presentación de la corrección de la base de datos

Resolución de preguntas

Pregunta: Obtenga el nombre de los discos que pertenecen a categoría de rock, que tengan un precio entre 10 y 20 euros
Respuesta: select titulo from album where genero='rock' and precio between(10 and 20);

Puntuación: 1

Comentario: ok!

Pregunta: Obtenga el nombre de los clientes que hayan comprado el disco American Idiot y hayan nacido antes de 1991
Respuesta: select nombre from clientes where id_cliente IN(select id_cliente from pedidos where id_pedido IN(select id_pedido from pedido_album where id_album IN(select id_album from album where titulo='American idiot')));

Puntuación: 1

Comentario: ok!

Pregunta: Elimina el album que valga más de 30 euros y pertenezca a la categoría Indie
Respuesta: delete from album where precio > 30 and genero='Indie';

Puntuación: 1

Comentario: ok!

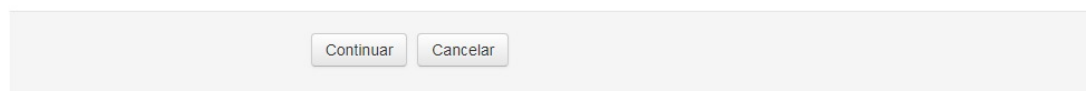


Ilustración 60: Presentación de la corrección de las respuestas

5.7. Correcciones

1. Para poder acceder a la sección de correcciones es necesario acceder con un perfil de administrador de moodle, profesor con privilegios o profesor sin privilegios.
2. Acceder a la instancia en la que se quiere evaluar.

La imagen muestra una interfaz de Moodle con un menú de actividades. En la parte superior, hay un ícono de notificación y el texto "Avisos" con un enlace de edición. Debajo, se ven tres secciones de actividades:

- Sección 1: "25 de febrero - 3 de marzo" con un ícono de plus y un enlace de edición.
- Sección 2: "4 de marzo - 10 de marzo" con un ícono de plus y un enlace de edición.
- Sección 3: "11 de marzo - 17 de marzo" con un ícono de plus y un enlace de edición.

Entre y después de cada sección, hay un botón que dice "+ Añade una actividad o un recurso".

Ilustración 61: Visualización del curso con rol de administrador o profesor

3. Acceder a la pestaña de puntuación situada en la parte superior derecha de la página

Ver base de datos Editar Base de datos Añadir preguntas Puntuación

Ilustración 62: Pestañas de navegación de la instancia

4. Una vez dentro, nos mostrará un listado con todos los alumnos que hay inscritos en el curso y el estado de realización de su ejercicio, si no lo ha completado aparecerá vacío, si lo ha completado y no está corregido aparecerá un radio button para que pueda ser elegido y si ya está calificado, aparecerá su calificación.

Ver base de datos Puntuación

Nombre apellido	Base de datos	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Evaluar
Pepito Lopez Parra							<input type="radio"/>
Luisito Perez Rodriguez							<input type="radio"/>
Manolito Sanchez Bermejo							<input type="radio"/>

Continuar Cancelar

Ilustración 63: Listado de alumnos que han realizado el ejercicio y se les puede evaluar

5. Para corregir a un alumno es necesario seleccionar el radio button que aparece al lado de su nombre. Cuando pulsemos continuar, nos llevará a una nueva página.

Ver base de datos Puntuación

Nombre apellido	Base de datos	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Evaluar
Pepito Lopez Parra							<input checked="" type="radio"/>
Luisito Perez Rodriguez							<input type="radio"/>
Manolito Sanchez Bermejo							<input type="radio"/>

Continuar Cancelar

Ilustración 64: Selección de alumno para evaluar

6. En esta página aparecerá la resolución completa del alumno y debajo de cada respuesta un cuadro de texto que solamente aceptará caracteres numéricos y otro cuadro de texto más grande para introducir los comentarios pertinentes por parte del profesor.

Resolución de base de datos

clientes

Nombre del campo	Primary key
admin	
fecha_nacimiento	
direccion	
codigo_postal	
email	
telefono	
id_cliente	Sí
apellidos	
nombre	
password	
dni	

album

Nombre del campo	Primary key
genero	
oferta	
id_album	Sí
caratula	
grupo	
ventas	
fecha_album	
precio	
lista_canciones	
titulo	

pedidos

Nombre del campo	Primary key
id_pedido	Sí
id_cliente	
coste_total	
fecha	

datospedido

Nombre del campo	Primary key
id_pedido	Sí
id_album	
cantidad	

Relaciones

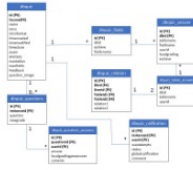
Nombre de la tabla 1	Tipo de relación tabla 1 -> tabla 2	Tipo de relación tabla 2 -> tabla 1	Nombre de la tabla 2
clientes	1	1..*	pedidos
album	1..*	1	pedidos
pedidos	1	1	datospedido

Puntuación*

Comentario

Ilustración 65: Presentación de la resolución del ejercicio con los cuadros de texto para añadir puntuación y comentarios

Resolución de preguntas



Obtenga el nombre de los discos que pertenecen a categoría de rock, que tengan un precio entre 10 y 20 euros

```
select titulo from album where genero='rock' and precio between(10 and 20);
```

Puntuación*

Comentario

Obtenga el nombre de los clientes que hayan comprado el disco American Idiot y hayan nacido antes de 1991

```
select nombre from clientes where id_cliente IN(select id_cliente from pedidos where id_pedido IN(select id_pedido from pedido_album where id_album IN(select id_album from album where titulo='American idiot')));
```

Puntuación*

Comentario

Elimina el album que valga más de 30 euros y pertenezca a la categoría Indie

```
delete from album where precio > 30 and genero='Indie';
```

Puntuación*

Comentario

Continuar

Cancelar

Ilustración 66: Presentación de las respuestas del alumno y los cuadros de texto para añadir puntuación y comentarios

- Una vez introducidos todos los campos si le damos a continuar volveremos a ver el listado de alumnos. En este caso, vemos que se ha actualizado la puntuación de ese alumno.

Ver base de datos	Puntuación	<input type="text"/>					
Nombre apellido	Base de datos	Pregunta 1	Pregunta 2	Pregunta 3	Pregunta 4	Pregunta 5	Evaluar
Pepito Lopez Parra	8	1	1	1	1	1	Puntuación total:...13
Luisito Perez Rodriguez							
Manolito Sanchez Bermejo							

Continuar Cancelar

Ilustración 67: Visualización de la puntuación que va a recibiendo cada alumno

6. Estudio económico

En este apartado vamos a hacer un recuento económico de lo que ha supuesto el desarrollo de este módulo. Para ello, contaremos con los útiles que se han empleado para el desarrollo y las horas de trabajo invertidas en el desarrollo:

Hardware:

- Se ha empleado un ordenador ASUS ZenBook con procesador i7-7500U, 16GB RAM y 512GB SSD. Este ordenador no se ha comprado exclusivamente para el desarrollo de este proyecto, sino que, ha sido empleado para otras funcionalidades. Así que diremos que su cómputo final para el presupuesto es una sexta parte de su valor. Originalmente, su precio es de 1500 €uros. Por tanto, su costo es de 250 €uros.

Software:

- Se ha empleado para alojar Moodle y el módulo un servidor local, WampServer. Esta aplicación contiene servidor SQL y PHP, entre otras cosas. Perfecto para el desarrollo que se ha realizado. Es de uso gratuito, por tanto, su costo final son 0 €uros.
- Para el desarrollo del código se ha empleado el procesador de código Sublime Text 2. Se trata de la versión gratuito de dicho software. Por tanto, coste final 0 €uros.
- El instalador de Moodle 3.2 para albergar nuestro módulo también es gratuito. Coste final 0 €uros.
- Template de plugin de Moodle, gratuito. Coste 0 €uros.

Capital humano:

- Supongamos que, para el desarrollo de la aplicación completo, se han empleado 240 horas. En la actualidad el salario por hora de un desarrollador backend / frontend ronda los 16,5 €uros. Por tanto, $240h \times 16,5 \frac{€}{h} = 3960 \text{ €uros}$.

Total:

- Hardware: 250 €uros.
- Software: 0 €uros.
- Capital Humano: 3960 €uros.
- Coste final: 4210 €uros

7. Conclusiones

En un mundo cada día más cambiante es imprescindible adaptar la metodología de estudio y la forma de impartir docencia a las nuevas tecnologías.

Con los avances tecnológicos que hoy en día dispone esta sociedad se ha roto definitivamente con la forma tradicional de transmitir los conocimientos. Se han eliminado las barreras físicas que impedían acceder a la información. Por tanto, es necesario aprovechar este impulso para que los alumnos y los docentes establezcan una relación mucho más profunda y enriquecedora. Las herramientas de las que puede disponer hoy en día un docente son mucho más adaptativas para todos los alumnos y le permiten perfeccionar más si cabe la forma de dar clase. Tanto el alumno como el profesor disponen del material y de los distintos recursos al instante en el punto geográfico que se encuentren y pueden acceder en el momento concreto que lo deseen. Esto permite que todo el mundo pueda organizar su estudio en función de sus circunstancias sin necesidad de tener que estar en un lugar determinado y en un momento concreto para ello.

Al entrar en juego los nuevos sistemas de información telemáticos, tanto alumnos como profesores son actores activos del aprendizaje. Para el profesor supone un gran avance en cuanto a la capacidad de difundir la información adaptándola a su forma de entenderla y así siendo más preciso a la hora de transmitirla. Por parte del alumno, se convierte en el epicentro de la actividad. El propio alumno es el que decide cómo y cuándo aprende, que aspectos del aprendizaje prefiere potenciar o en qué punto de la materia prefiere hacer foco. Se trata de un aprendizaje interactivo que sumerge de lleno al alumno en la materia y le hace participe en todo momento de ella.

Se ha desarrollado un módulo de gran utilidad para los profesores y los alumnos, con gran interactividad y que permitirá a todos hacer más completo el aprendizaje.

Los aspectos importantes a destacar sobre este proyecto son los siguientes:

- Al crear este proyecto basado en e-learning se ha mantenido un contacto profundo con esta forma de enseñanza y se ha aprendido sobre ella.
- Al comparar gran cantidad de plataformas LMS, se ha podido entender cuál es la naturaleza de estas y el gran beneficio que supone su utilización.
- Se ha puesto en valor la utilización de estas plataformas y el porqué de su masiva utilización en todo tipo de institución y empresas.
- Gracias al estudio en profundidad de la herramienta moodle se ha podido conocer teorías educativas, como el constructivismo, muy interesantes que enriquecen el conocimiento y que permiten acercarse a otros ámbitos del conocimiento propios de otras ramas.
- La revisión de las tecnologías, sobre las que moodle está desarrollado, ha permitido el conocimiento más en profundidad de lenguajes que actualmente están en vanguardia. Permitiendo así estar al día de las últimas herramientas de desarrollo.
- Desarrollando el módulo dbquiz ha permitido conocer, desde una óptica distinta a la habitual, todos los procesos necesarios para desarrollar y producir una herramienta funcional. Muy parecido a lo que se realiza en un puesto de desarrollador.

- La creación del módulo ha permitido conocer el trabajo docente. Teniendo que pensar y ajustar el desarrollo a las necesidades del docente. Desde la creación del ejercicio hasta la calificación de los alumnos, pasando por los plazos de entrega.
- El módulo, basado principalmente en la adquisición de conocimiento en la creación de base de datos y en las consultas a las mismas, ha permitido en sí mismo, la adquisición de mayores conocimientos en la creación de bases de datos y de las consultas. También en conocer más de cerca el lenguaje MySQL.

8. Líneas futuras

En la siguiente sección vamos a tratar posibles mejoras o avances a realizar en el módulo dbquiz. Si bien el desarrollo y despliegue del módulo cumple con los requisitos solicitados con una estructura sólida y bien fundamentada capaz de dar soporte en asignaturas con Laboratorio de desarrollo de sistemas telemáticos (LDST) o Tecnologías para aplicaciones web (TAW), es importante destacar posibles puntos de mejora con los que se podría continuar trabajando con el fin de disponer de una herramienta versátil y puntera.

Este módulo, como ya hemos comentado sobradamente, sirve para entrenarse en la creación de las bases de datos y de creación de consultas a base de datos MySQL.

A continuación, veremos las posibles mejoras a desarrollar:

8.1. Edición de preguntas

En la actual versión del módulo dbquiz la edición de preguntas se realiza a través de la edición de la base de datos. Esto supone que deberemos eliminar la base de datos creada y con ella serán eliminadas las preguntas. Esto supone un inconveniente a la hora de editar una pregunta por el tedioso procedimiento que supone. En este caso, como solución de mejora se propone la creación de una nueva pestaña de edición de preguntas en la que las preguntas se puedan seleccionar a una y acometer los cambios necesarios para posteriormente ser guardados.

8.2. Resolución del ejercicio

- Resolución de la base de datos:

En la resolución del ejercicio se propone explorar opciones más interactivas, basadas en nuevas soluciones de JavaScript en las que el alumno podría dibujar sobre un lienzo su propia base de datos, o bien una solución que a través de módulos de construir pueda crear su base de datos como si se tratase de un puzle. Esto se viene empleando en nuevas tecnologías, como en los entornos de desarrollo orientados niños conocido como scratch. En el cual el usuario dispone de bloques, los cuales tienen cada uno su función. Según la colocación de los mismos podrá formar su programa. Si lo trasladamos a nuestro campo, el alumno podría construir la base de datos con bloques customizados por el mismo.

Esta idea viene a resaltar la idea de la libertad de creación por parte del alumno permitiendo ampliar sus posibilidades de aprender y estimulando su imaginación.

- Feedback:

En la actual configuración el feedback aportado por el profesor solo aparece en caso de que el alumno solicite volver a realizar el ejercicio. De modo que, para recibir la ayuda en alumno debe completar el ejercicio entero. Además, se trata de un feedback muy genérico, donde en un cuadro de texto el profesor debe incluir todas las pistas juntas. Esta línea de mejora tiene como objetivo ayudar lo máximo posible al alumno. Al igual que ahora, cuando la instancia creada se trate de un examen el feedback desaparecería.

Debería de tratarse de un feedback flexible que pueda ser customizado por el profesor y que pudiera ser colocado en lugares estratégicos donde el profesor necesite reforzar algún conocimiento. También debería permitir añadir distintos ítems para que cada uno de ellos fuese colocado en un lugar concreto y que, cuando el alumno lo desee, pueda acceder a ello durante la realización del ejercicio.

Por otro lado, sería interesante que el feedback no se limitase a texto plano como tal. Si bien podría incluir imagen o vídeos explicativos que favorezcan el entendimiento del ejercicio.

8.3. Calificaciones

En el apartado de las calificaciones las herramientas de las que dispone el profesor son un cuadro de texto para introducir la puntuación y un cuadro para incluir comentarios. A todas luces, esto se queda ligeramente escueto a la hora de dar una respuesta clara y concisa al alumno sobre su ejercicio. No pudiendo dar una visión exacta de los puntos a mejorar.

La propuesta de mejora se basa en que el profesor pueda interaccionar directamente con la base de datos propuesta por el alumno. Realizando cambios o añadiendo notas. Como si se tratase de una pizarra o un folio. A la hora de que el alumno revise su calificación le será más útil en su aprendizaje.

8.4. Sección de estadísticas

Sería muy interesante habilitar una sección de estadísticas en la que se registrase la actividad de una instancia. Los fallos más recurrentes, aciertos, etc. Esto permitiría al profesor entender cuáles son los puntos que debe profundizar para el conjunto de la clase y también ver la situación real del aprendizaje el curso.

En global, también sería bueno tener estadísticas. Tanto para el profesor como el alumno, es necesario conocer su progreso y poder descubrir cuáles son los puntos fuertes y débiles.

Si en el apartado de preguntas se utilizasen tags para categorizar las preguntas, estas podrían ser medidas y en global estudiar los apartados a mejorar.

Esta mejora supondría un salto de calidad importante para el módulo porque se trataría de un seguimiento exhaustivo del estudio permitiendo a todos ser más eficientes.

8.5. Concepto de sandbox

A día de hoy, es conocido la capacidad de a través de inyección código SQL permite hackear cualquier sistema. Si desde este módulo se diese cualquier posibilidad a que alguien con los conocimientos necesarios acceder directamente a la base de datos de moodle sería una falla de seguridad muy seria que comprometería de forma grave la integridad de la propia plataforma. Por esta razón no se ha habilitado la ejecución de las respuestas de los alumnos para la corrección.

En este punto sería muy útil crear un control sobre las consultas que se pudieran hacer. Como si de un cajón de arena en el que un niño puede jugar sin hacer daño. El sandbox permitiría ejecutar dichas consultas con la suficiente seguridad. Esto permitiría al alumno tener una respuesta al instante lo que conseguiría dar más interacción entre el alumno y la base de datos.

9. Referencias

- alfredotemiquel. (17 de Agosto de 2017). *MDN web docs Mozilla*. Obtenido de MDN web docs Mozilla: <https://developer.mozilla.org/es/docs/HTML/HTML5>
- Alhambra-eidos. (2012). *Almagesto elearning Management System*. Obtenido de Almagesto elearning Management System: <http://www.almagesto.com/>
- Anónimo. (15 de Agosto de 2017). *Librosweb*. Obtenido de Librosweb: http://librosweb.es/libro/xhtml/capitulo_1/breve_historia_de_html.html
- Anónimo. (11 de Agosto de 2017). *Universitat Oberta de Catalunya*. Obtenido de Universitat Oberta de Catalunya: <http://cv.uoc.edu/web/~fcanoco/img/CSS.pdf>
- apereo Foundation. (2014). *Sakai*. Obtenido de <https://www.sakaiproject.org/>
- Chamilo. (2014). *Chamilo Elearning & Collaboration software*. Obtenido de Chamilo Elearning & Collaboration software: <https://chamilo.org/es/chamilo-lms/>
- Cruz Del Valle, E. (2017). *EJECUTIVOS de INFORMÁTICA - TEMARIO MATERIAS ESPECÍFICAS*. Zaragoza.
- educativa. (2018). *educativa*. Obtenido de educativa: <https://www.educativa.com/>
- Groff, J., Weinberg, P., & Opper, A. (2010). *SQL: The Complete Reference, Third Edition*. London: McGraw-Hill.
- Inc., B. (2018). *Blackboard*. Obtenido de <http://es.blackboard.com/>
- jlopezfdez. (11 de Agosto de 2017). *MDN webs docs mozilla*. Obtenido de MDN webs docs mozilla: <https://developer.mozilla.org/es/docs/Web/JavaScript/Guide/Introduccion>
- Mariel Castro, S., Ariel Clarenc, C., López de Lenz, C., Moreno, M. E., & Tosco, N. B. (2013). *Analizamos 19 plataformas de eLearning: Investigación colaborativa sobre LMS*. Grupo GEIPITE.
- Moodle. (9 de Agosto de 2017). *Moodle Docs*. Obtenido de Moodle: https://docs.moodle.org/all/es/Herramientas_para_desarrolladores
- Moodle: Características de Moodle 3.3. (Agosto de 2017). *Moodle*. Obtenido de Moodle: https://docs.moodle.org/all/es/Caracteristicas_de_Moodle_3.3
- Moodle: Gestionando un sitio Moodle. (Agosto de 2017). *Moodle*. Obtenido de Moodle: https://docs.moodle.org/all/es/Gestionando_un_sitio_Moodle
- Moodle: Managing a Moodle course. (Agosto de 2017). *Moodle*. Obtenido de Moodle: https://docs.moodle.org/34/en/Managing_a_Moodle_course
- Moodle: Moodle Mobile features. (Agosto de 2017). *Moodle*. Obtenido de Moodle: https://docs.moodle.org/34/en/Moodle_Mobile_features

Moodle: NEWMODULE Documentation. (Agosto de 2017). *Moodle*. Obtenido de Moodle:
https://docs.moodle.org/dev/NEWMODULE_Documentation

Schools, w. (11 de Enero de 2018). *w3c Schools*. Obtenido de w3c Schools:
<https://www.w3schools.com/w3css/>

The PHP Group. (10 de Agosto de 2017). *PHP*. Obtenido de PHP:
<http://php.net/manual/es/history.php.php>

The PHP Group. (10 de Agosto de 2017). *PHP*. Obtenido de PHP:
<http://php.net/manual/es/intro-whatcando.php>

UNESCO. (24 de Septiembre de 2014). *United Nations Educational, Scientific and Cultural Organization*. Obtenido de UNESCO Web site: <https://es.unesco.org/news/medio-mundo-estar%C3%A1-l%C3%ADnea-2017>

Valdés, D. P. (3 de Julio de 2007). *Maestro del web by platzi*. Obtenido de Maestro del web by platzi: <http://www.maestrosdelweb.com/que-es-javascript/>