

UNIVERSIDAD DE



VALLADOLID

E.T.S.I. TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

GRADO EN INGENIERÍA DE TECNOLOGÍAS DE TELECOMUNICACIÓN

ADAPTACIÓN DE LA GESTIÓN DE GRUPOS DEL MÓDULO BLENDED 1.0 A LA VERSIÓN DE MOODLE 2.5

Autor:

Dña. Jéssica Olano López

Tutor:

D. Juan Pablo de Castro Fernández

Valladolid, 05 de Septiembre de 2014

TÍTULO: Adaptación de la gestión de grupos del módulo
Blended 1.0 a la versión de Moodle 2.5

AUTOR: Dña. Jéssica Olano López

TUTOR: D. Juan Pablo de Castro Fernández

DEPARTAMENTO: Departamento de Teoría de la señal y
Comunicaciones e Ingeniería Telemática.

TRIBUNAL

PRESIDENTE: D. Juan Pablo de Castro Fernández

VOCAL: Dña. Luisa María Regueras Santos

SECRETARIO: Dña. María Jesús Verdú Pérez

SUPLENTE: D. Jaime Gómez Gil

SUPLENTE: D. Federico Merino Caminero

FECHA: 05 de Septiembre de 2014

CALIFICACIÓN:

Resumen de TFG

El presente trabajo engloba el desarrollo de una aplicación ligada a las tecnologías de teleformación o *e-learning*, que son aquellas que proporcionan un espacio virtual de aprendizaje orientado a facilitar la experiencia de una formación a distancia. Este trabajo está basado en el desarrollo y adaptación de un módulo desarrollado previamente para la plataforma de teleformación *Moodle*, cuya implantación y uso está cada vez más extendido en la Universidad de Valladolid. En este trabajo se produce la adaptación del módulo *Blended* implementado para versiones anteriores de *Moodle* y en concreto, se desarrolla y actualiza a una nueva versión de *Moodle* del módulo en el que se implementan la creación y gestión de agrupamientos como método para la creación de equipos en cualquier tarea de una manera más eficiente a la utilizada con anterioridad. Gracias a esta nueva adaptación, será más sencilla la interacción entre alumnos y profesores y se conseguirá un aprendizaje más eficiente y cercano.

Palabras clave

Plataforma *Moodle*, módulo *Blended*, *PHP*, *mySQL*, *JavaScript*, agrupamientos, equipos, calificación y actividad.

Abstract

This work includes the development of an application linked to learning technologies or *e-learning* that would provide a virtual learning space designed to facilitate a distance learning experience. This work is based on the development and adaptation of a previously developed module for *Moodle* learning platform, whose implementation and usage is increasingly extended at the University of Valladolid. This work presents the adaptation of *Blended* module implemented for older *Moodle* versions, and in particular, is developed and updated to a new module Moodle version in which the creation and management of groupings are implemented like a method to create teams for any *Moodle* task in a more efficient way that used previously. With this new adaptation will be easier the interaction between students and teachers and it will achieve a more efficient and nearby learning.

Keywords

Moodle platform, *Blended* module, *PHP*, *MySQL*, *JavaScript*, groupings, groups, grade, activity.

Agradecimientos

Quiero dedicar este trabajo a mis padres Jose Eloy y Aurora y a mis abuelos Eloy y Rosa por haberme apoyado durante estos años y haber confiado en mí.



Esta obra está bajo una [licencia de Creative Commons](#).

Tabla de Contenidos

Tabla de Contenidos	I
Índice de tablas	III
Índice de figuras	V
Parte 1. Introducción y conocimientos previos	1
Capítulo 1. Introducción, motivación y objetivos	2
1.1. Introducción	2
1.2. Motivación	2
1.3. Objetivos	3
Capítulo 2. Conocimientos previos	5
2.1. Plataforma Moodle.....	5
2.1.1. <i>Propiedades</i>	6
2.1.2. <i>Características</i>	7
Capítulo 3. Descripción del problema.....	10
3.1. Introducción	10
3.2. Requisitos del proyecto	10
3.2.1. <i>Requisitos funcionales del módulo</i>	10
3.2.2. <i>Requisitos técnicos de Moodle</i>	13
Parte 2. Trabajo desarrollado	15
Capítulo 4. Teoría	16
4.1. Introducción	16
4.2. Esquema de un módulo	16
4.3. Diseño de la base de datos	18
4.3.1. <i>Tablas</i>	18
4.3.2. <i>Diagrama Entidad - Relación</i>	25
4.4. Ficheros del módulo.....	27
4.4.1. <i>Ficheros generales del módulo</i>	27
4.4.2. <i>Ficheros de Blended</i>	50
Capítulo 5. Análisis de resultados: manual de usuario.....	68
5.1. Instalación y desinstalación de <i>Blended</i>	68

5.1.1. Instalación	68
5.1.2. Desinstalación	69
5.2. Configuración de Blended.....	69
5.3. Manejo de <i>Blended</i>	71
5.3.1. <i>Páginas disponibles tanto a profesores como alumnos</i>	71
5.3.1.1. Página principal de Blended.....	71
5.3.1.2. Generar página de etiquetas	74
5.3.1.3. Generar página de tarea.....	77
5.3.2. <i>Páginas disponibles únicamente a profesores</i>	79
5.3.2.1. Administración de equipos.....	79
5.3.2.2. Calificar tareas	83
5.3.3. <i>Páginas disponibles únicamente a alumnos</i>	85
5.3.3.1. Inscribirse en equipo	85
5.4. Copia de seguridad.....	93
Parte 3. Conclusiones.....	95
Capítulo 6. Conclusiones	96
Capítulo 7. Líneas futuras	97
Bibliografía y referencias	99

Índice de tablas

Tabla 1: Tabla <i>blended</i>	19
Tabla 2: Tabla <i>blended_team</i>	19
Tabla 3: Tabla <i>blended_member</i>	20
Tabla 4: Tabla <i>blended_grade</i>	20
Tabla 5: Tabla <i>blended_assign_grouping</i>	21
Tabla 6: Tabla <i>groupings</i>	21
Tabla 7: Tabla <i>groupings_groups</i>	22
Tabla 8: Tabla <i>groups</i>	22
Tabla 9: Tabla <i>groups_members</i>	23
Tabla 10: Tabla <i>grade_items</i>	24
Tabla 11: Tabla <i>grade_grades</i>	25
Tabla 12: Diagrama entidad-relación	25
Tabla 13: Función <i>blended_add_instance</i>	33
Tabla 14: Función <i>blended_update_instance</i>	33
Tabla 15: Función <i>blended_delete_instance</i>	34
Tabla 16: Función <i>blended_gen_cleanidvalue</i>	34
Tabla 17: Función <i>blended_clean_idnumber</i>	34
Tabla 18: Función <i>blended_update_grouping</i>	35
Tabla 19: Función <i>blended_update_group</i>	35
Tabla 20: Función <i>blended_add_new_group</i>	36
Tabla 21: Función <i>blended_method_random</i>	36
Tabla 22: Función <i>blended_get_course_students_ids</i>	37
Tabla 23: Función <i>blended_delete_groupings</i>	37
Tabla 24: Función <i>blended_actualizar_agrupamiento</i>	38
Tabla 25: Función <i>blended_probe_group_exist</i>	39
Tabla 26: Función <i>blended_update_teams_by_groups</i>	39
Tabla 27: Función <i>blended_num_members</i>	40
Tabla 28: Función <i>blended_insert_teams</i>	41
Tabla 29: Función <i>blended_insert_team</i>	42
Tabla 30: Función <i>blended_delete_sign_member_or_group</i>	42
Tabla 31: Función <i>blended_comprobar_calificacion</i>	43
Tabla 32: Función <i>blended_is_not_active_student</i>	43
Tabla 33: Función <i>blended_get_teams_members</i>	44
Tabla 34: Función <i>blended_get_teams_members_from_grades</i>	45
Tabla 35: Función <i>blended_get_name_team</i>	45
Tabla 36: Función <i>blended_get_assignment_name</i>	46
Tabla 37: Función <i>blended_get_user</i>	46
Tabla 38: Función <i>blended_get_grades</i>	47
Tabla 39: Función <i>blended_count_repeat_values</i>	47
Tabla 40: Función <i>blended_get_grademax</i>	47
Tabla 41: Función <i>blended_get_error_alert</i>	48

Tabla de contenidos

Tabla 42: Función <i>pdfAssignmentPage</i>	48
Tabla 43: Función <i>blended_crear_equipo_por_alumno</i>	49
Tabla 44: Función <i>blended_crear_nuevo_agrupamiento_por_alumno</i>	49
Tabla 45: Función <i>blended_inscribir_equipo_por_alumno</i>	50

Índice de figuras

Figura 1: Instalación de <i>Blended</i>	68
Figura 2: Desinstalación de <i>Blended</i>	69
Figura 3: Configuración de <i>Blended</i>	71
Figura 4: Vínculos visibles al profesor	72
Figura 5: Vínculos visibles al alumno	73
Figura 6: Generar página de etiquetas por alumno	74
Figura 7: Generar página de etiquetas por profesor	75
Figura 8: Etiquetas EAN13	76
Figura 9: Etiquetas QR2D	76
Figura 10: Mensaje de alerta generación de etiquetas	77
Figura 11: Generar página de tarea por alumno	77
Figura 12: Generar página de tarea por profesor	78
Figura 13: Ejemplo página de tarea	78
Figura 14: Mensaje de alerta generación de página de tarea	79
Figura 15: Administración de equipos	79
Figura 16: Inicializar equipos	80
Figura 17: Elegir agrupamiento	81
Figura 18: Nuevo agrupamiento	81
Figura 19: Modificar equipos	82
Figura 20: Equipos con identificador repetido	83
Figura 21: Calificar tareas	84
Figura 22: Introducir calificaciones	84
Figura 23: Alerta de calificación individual	85
Figura 24: Inscribirse en equipo	86
Figura 25: Mensaje de denegación de la inscripción	87
Figura 26: Alumno ya inscrito en un equipo	87
Figura 27: Ejemplo inscribirse en un equipo	88
Figura 28: Ejemplo estudiante inscrito	89
Figura 29: Creación de agrupamiento por alumno	89
Figura 30: Agrupamiento y grupo creado por alumno	90
Figura 31: El alumno puede crear un equipo	91
Figura 32: Alumno crea un nuevo equipo	91
Figura 33: Equipo con líder	92
Figura 34: Acceso a copia de seguridad	93

Parte 1. Introducción y conocimientos previos

Capítulo 1. Introducción, motivación y objetivos

1.1. Introducción

Este trabajo de fin de grado del Grado en Ingeniería de Tecnologías de Telecomunicación implementa el desarrollo y actualización del módulo Blended, creado para una versión antigua de la plataforma de teleformación de código abierto *Moodle*, a una nueva versión mejorada de éste, contribuyendo a ser una plataforma con una inmensa funcionalidad y aplicación en el ámbito académico tanto para profesores como para alumnos. Su uso en la Universidad introduce una nueva forma de aprendizaje y enseñanza mediante la web.

En este trabajo presentado, se persigue la adaptación de un módulo en concreto que ha sido desarrollado para las nuevas versiones de *Moodle*: los agrupamientos. Se trata de una herramienta de creación de equipos que permite a los profesores la agrupación de los alumnos para cualquier actividad presente en *Moodle*.

Con la implementación de la versión 2.x de *Moodle* se ofreció un soporte que permite la creación de grupos de trabajo que los profesores pueden usar para crear grupos de alumnos y asociarlos a una de las tareas propuestas definidas en *Moodle*. A su vez, los agrupamientos permiten definir un conjunto de grupos, de esta manera se pueden asociar distintos grupos a una misma tarea sin más que asociar el agrupamiento a dicha tarea.

Mediante la nueva versión de Blended, se permitirá tanto la creación de agrupamientos como de grupos de trabajo de una manera más sencilla e intuitiva a la proporcionada con *Moodle*, facilitando el uso de esta nueva herramienta desarrollada.

1.2. Motivación

Con los avances tecnológicos sufridos en los últimos años, la utilización de una plataforma que permita facilitar el trabajo de un aula de manera virtual supone un gran apoyo a la docencia presencial. *Blended* es un módulo que facilita la interacción entre profesores y alumnos mediante una serie de funciones que permiten

no sólo integrar actividades evaluables si no poder crear grupos de trabajo para de esta manera poder agrupar a los alumnos de manera virtual, rápida y segura.

1.3. Objetivos

Los objetivos que persigue este trabajo son los siguientes:

- ***Agrupación de alumnos por equipos:*** A la hora de realizar el desarrollo de diversas actividades que pueden ser propuestas en *Moodle* un profesor puede requerir la creación de diversos equipos de trabajo asociados a la actividad que desee. Mediante la creación de agrupamientos que recojan los grupos creados, se permite de manera sencilla puntuar los grupos sin necesidad de calificar alumno por alumno.
 - ***Asignación de agrupamiento:*** un agrupamiento puede ser creado mediante la interfaz dada por *Moodle* o creado de manera manual por un profesor. Los grupos de éste, pueden ser también creados con *Moodle* o elegidos por el profesor a la hora de crear el agrupamiento. Un alumno también tiene la posibilidad, si el profesor lo permite, de poder crear un grupo dentro de un agrupamiento existente o incluso crear un nuevo agrupamiento.
Si son creados por el profesor, elegirá el número de equipos que desea crear y cuantos miembros tendrá cada uno. Los equipos se pueden crear de dos maneras:
 - Manual: El profesor introduce de manera manual los alumnos en los grupos introduciendo su código de identificación, que en este módulo es representado por el DNI sin la letra.
 - Aleatorio: Los alumnos son introducidos de manera aleatoria y automática en los grupos. El profesor tiene la capacidad de elegir si desea introducir a todos los alumnos matriculados o solo los que estén activos en el curso.
- ***Administración de los equipos de un agrupamiento:*** Una vez asociado un agrupamientos con sus respectivos grupos a una actividad, el profesor tiene la capacidad de modificarlos de distintas formas:
 - ***Modificación de los alumnos:*** El profesor puede modificar los grupos introduciendo alumnos, un nuevo grupo o eliminando alumnos o grupos.
 - ***Re-inicialización de los grupos del agrupamiento:*** Si se desea, se pueden volver a introducir desde cero los alumnos y los grupos

eligiendo el número de equipos, miembros por equipo y el tipo de creación que se desea realizar: manual o aleatoria.

- ***Re-asociación del agrupamiento:*** El profesor tiene la capacidad de cambiar el agrupamiento asociado a una tarea concreta, bien sea eligiendo uno existente y por tanto asignado los grupos que éste posea a la tarea, o creando un nuevo agrupamiento con sus respectivos grupos.

Si es un alumno el que ha creado el grupo, puede tener la posibilidad de poder eliminar los miembros del grupo que desee con los que no esté conforme siempre y cuando el profesor le conceda ese tipo de permiso.

- ***Calificación de los equipos:*** Una vez asociados equipos a una tarea, un profesor tiene la posibilidad de calificar los grupos que ésta posea.

Los detalles de cada uno de estos objetivos se desarrollan en el capítulo 3 (*Análisis del problema*) de esta memoria.

Capítulo 2. Conocimientos previos

2.1. Plataforma Moodle

Moodle es un sistema de gestión de aprendizaje online y gratuito que permite a los educadores crear su propio sitio web privado lleno de cursos dinámicos que extienden el aprendizaje, en cualquier momento y en cualquier lugar. *Moodle* es capaz de satisfacer las necesidades tanto de profesores como de alumnos gracias a la cantidad de características que posee.

Moodle fue creado por Martin Dougiamas, nacido en Australia a finales de 1970, estudiante de la Universidad de Curtin, trabajó con WebCT lo que le llevó a investigar un proyecto para la enseñanza en línea, desarrollando sus primeros prototipos en 1999.

La primera versión de *Moodle* fue lanzada en Agosto de 2002, creciendo rápidamente lo que le llevó a convertirse en 2007 en empresa líder. En 2010, *Moodle* estaba compuesto por más de un millón de usuarios y más de 50 socios. La nueva versión de *Moodle* 2.0 se lanzó en Noviembre de 2010 tras la cual han ido apareciendo sucesivas actualizaciones de la plataforma.

La palabra *Moodle* es un acrónimo de *Modular Object-Oriented Dynamic Learning Environment* (Entorno Modular de Aprendizaje Dinámico Orientado a Objetos), que es sobre todo útil para programadores y teóricos de la educación. También es un verbo anglosajón que describe el proceso de deambular perezosamente a través de algo, hacer las cosas como se te ocurre hacerlas, de una forma que a menudo lleva a la visión y la creatividad. Como tal, se aplica tanto a la forma en que *Moodle* se desarrolló originalmente como a la manera en que un estudiante o profesor podría aproximarse al estudio o enseñanza de un curso en línea de manera permanente e iterativa.

La instalación requiere una plataforma soporte sobre *PHP* además de la necesidad de una base de datos *SQL* como por ejemplo *MySQL*. La licencia que utiliza *Moodle* es la *GNU General Public License* [1].

2.1.1. Propiedades

Facilidad de uso

Una interfaz sencilla, funciones de arrastrar y soltar, y recursos bien documentados que junto con mejoras de usabilidad en curso hacen a *Moodle* fácil de aprender y usar.

Código abierto

Moodle se distribuye gratuitamente como *Open Source software*, bajo la Licencia Pública General de GNU. Cualquiera puede adaptar, ampliar o modificar *Moodle* para proyectos comerciales y no comerciales, sin ningún tipo de derechos de licencia y beneficiarse de las eficiencias de costes, flexibilidad y otras ventajas del uso de *Moodle*.

Numerosos recursos disponibles

Acceso de manera extensa a la documentación de Moodle y foros de usuarios en varios idiomas, libre de contenido y cursos compartidos por los usuarios de *Moodle* en todo el mundo, así como cientos de *plugins* y complementos aportados por una gran comunidad global.

Moodle está basado en web y por lo tanto se puede acceder desde cualquier parte del mundo. El contenido en la plataforma *Moodle* es fácilmente accesible y consistente a través de diferentes navegadores web y dispositivos.

Escalable

Desde unos pocos estudiantes a millones de usuarios, *Moodle* se puede escalar para soportar las necesidades de ambas clases pequeñas y grandes organizaciones.

La característica de escalabilidad que *Moodle* posee, se ve reflejada principalmente en los usuarios que lo utilizan. Gracias a su licencia *Open Source*, el

ser una plataforma gratuita hace que los usuarios vayan desde universidades y escuelas a instituciones militares y aéreas.

Robusto, seguro y privado

La infraestructura de *Moodle* puede soportar de manera confiable y efectiva una implementación productiva a gran escala las 24 horas del día, los 7 días de la semana.

A pesar de ser una plataforma gratuita que cualquier usuario puede poseer, *Moodle* posee una alta seguridad contra accesos no autorizados y se asegura de evitar el uso indebido de la plataforma.

Todos los datos almacenados en la plataforma son protegidos lo que hace de *Moodle* un sitio seguro y privado donde las comprobaciones de seguridad son prioritarias.

Interoperabilidad

En lo referente a la autenticación *Moodle* utiliza como método principal el protocolo *LDAP* aunque también tiene la posibilidad de utilizar otro tipo de métodos estándar como *IMAP*.

En la matriculación *Moodle* permite la utilización de un servidor *LDAP* y el estándar *IMS Enterprise*.

Con respecto a la importación/exportación de información *Moodle* utiliza principalmente el estándar *XML*.

2.1.2. Características

Características generales

- Interfaz fácil de usar: la interfaz de *Moodle* es fácil de navegar tanto con dispositivos de escritorio como móviles.

- Tablón de actividades personalizable: el usuario tiene la posibilidad de organizar y mostrar los cursos de la forma que desee. pudiendo echar un vistazo a las tareas y mensajes recientes.
- Herramientas y actividades de colaboración: *Moodle* da la posibilidad de trabajar y aprender de manera conjunta en foros, wikis, glosarios, actividades de base de datos, etc.
- Calendario: La herramienta de calendario de *Moodle* ayuda a mantener un seguimiento del calendario académico o de la empresa, los plazos de los cursos, reuniones de grupo, etc.
- Gestión de archivos: capacidad para incluir o eliminar archivos de los servicios de almacenamiento en nube, incluyendo Skydrive, Dropbox y Google Drive.
- Editor de texto: *Moodle* permite dar formato al texto y agregar de manera adecuada elementos multimedia e imágenes con un editor que es capaz de funcionar en todos los navegadores y dispositivos.
- Notificaciones: Si el servicio de notificaciones se encuentra activo, los usuarios pueden recibir alertas sobre nuevas tareas y plazos, mensajes en el foro y también enviarse mensajes privados entre sí.
- Control del progreso: los profesores y los alumnos pueden seguir el progreso y la finalización de las actividades o recursos tanto individuales como del curso.

Administración del sitio

- Diseño del sitio adaptable: *Moodle* da la posibilidad al usuario de personalizar de manera sencilla un “tema” de *Moodle* con un logotipo, colores,... además de poder diseñar su propio “tema”.
- Autenticación segura: Más de 50 maneras de autenticación y matriculación para agregar e inscribir a los usuarios.
- Capacidad multilingüe: Capacidad de permitir a los usuarios ver el contenido del curso en diferentes idiomas siempre y cuando haya sido configurados previamente.
- Creación de cursos y copias de seguridad: capacidad para añadir cursos manualmente, realizar una copia de seguridad o realizar restauraciones.
- Administrar roles y permisos de usuario: capacidad de definición de roles para poder conocer qué tipo de usuarios acceden al curso.
- Soporta estándares abiertos.
- Actualizaciones de seguridad regulares: *Moodle* se actualiza regularmente con los últimos parches de seguridad.

- Generación de informes detallados sobre la actividad y la participación en el curso.

Administración de cursos

- Itinerarios de aprendizaje directos: Posibilidad de diseñar y gestionar cursos de la manera que el usuario desee. *Moodle* proporciona la posibilidad de que las clases sean dirigidas por un instructor, autodidácticas o en línea.
- Ofrece una serie de actividades para los cursos: foros, glosarios, cuestionarios, recursos, consultas, encuestas, tareas, chats y talleres.
- Integración de los recursos externos: capacidad por parte del profesorado de incorporar materiales en el curso e introducir calificaciones.
- Integración multimedia: *Moodle* posee un soporte integrado de medios que le permite buscar e insertar de manera sencilla archivos de audio y video en sus cursos.
- Dirección del grupo: Los grupos de alumnos pueden compartir recursos, diferenciar actividades y facilitar el trabajo en equipo.
- La mayoría de las áreas para introducir texto (recursos, envío de mensajes a un foro, etc.) pueden editarse usando un editor *HTML* integrado.

Administración de usuarios

- El objetivo principal es reducir al mínimo el trabajo del administrador, manteniendo un alto nivel de seguridad en la plataforma.
- Método estándar de alta por correo electrónico: los estudiantes pueden crear sus propias cuentas de acceso. La dirección de correo electrónico se verifica mediante confirmación.
- Método *LDAP*: las cuentas de acceso pueden verificarse en un servidor *LDAP*. El administrador puede especificar qué campos usar.
- *IMAP, POP3, NNTP*: las cuentas de acceso se verifican con un servidor de correo o de noticias (*news*). Soporta los certificados *SSL* y *TLS*.
- Base de datos externa: cualquier base de datos que contenga al menos dos campos puede usarse como fuente externa de autenticación.
- Una cuenta de administrador controla la creación de cursos y determina los profesores, asignando usuarios a los cursos.
- A los profesores se les puede borrar los privilegios de edición para que no puedan modificar el curso (e.j. para tutores a tiempo parcial).
- Cada usuario puede elegir el idioma que se usará en la interfaz de *Moodle* (Inglés, Francés, Alemán, Español, Portugués, etc.).

Capítulo 3. Descripción del problema

3.1. Introducción

Al iniciar este trabajo, la primera tarea realizada ha sido la especificación de los requisitos que el módulo debería cumplimentar. Gracias a las sucesivas entrevistas con el tutor, el proyecto se dividió en dos partes principales:

1. *Adaptación* del módulo Blended creado para *Moodle 1.x* a la versión actualizada de *Moodle 2.x*.
2. Integración con la nueva función implantada en *Moodle 2.x* de la posibilidad de la realización de agrupamientos. Se dispondrá de uso de forma más simplificada a la propuesta por *Moodle*.

Tras haber cumplimentado los requisitos especificados, se procedió a la validación de éstos para comprobar que el funcionamiento se ajustaba a las especificaciones pedidas.

3.2. Requisitos del proyecto

3.2.1. Requisitos funcionales del módulo

Identificación de los alumnos

- Cada alumno introducido en el sistema tendrá un código de identificación único. El tipo de código que se introducirá será configurado en las características de configuración del módulo pudiendo elegir entre diversas opciones como id de *Moodle*, id de usuario, id personalizado, etc.
- Un profesor podrá imprimir páginas de etiquetas formadas por un código de barras con las que se podrá identificar a cada alumno. Un alumno también tiene la posibilidad de imprimir sus propias etiquetas.

En esta función se pueden fijar el número de etiquetas por fila y por columna para ajustar el tamaño del código de barras a las etiquetas adhesivas que se vayan a utilizar.

- Un profesor también tendrá la posibilidad de generar páginas de tarea que contendrán la información de ésta (nombre, fecha de entrega y descripción), además del código de barras de cualquier alumno que desee. El alumno tiene la posibilidad de poder imprimir sus páginas de tarea.

Estas dos últimas funciones planteadas sólo se podrán llevar a cabo si el alumno está matriculado y posee código de identificación en el sistema. Si esto no se ha realizado, se le mostrará un mensaje de alerta al profesor indicándole la razón por la cual la función elegida no puede ser llevada a cabo.

Administración de equipos

- Profesor:

Para llevar a cabo la administración de equipos, se mostrará al profesor una tabla con todas las actividades calificables que han sido generadas en el curso y en las que se puedan introducir grupos de trabajo. Se mostrará el nombre de la actividad que será un enlace a la página de dicha actividad, su fecha de entrega (si posee) y para la administración de los equipos se mostrarán tres enlaces:

- Inicializar equipos: Con este botón se podrá asignar un agrupamiento a la actividad elegida. En este caso hay dos posibilidades: elegir un agrupamiento existente con sus grupos o crear un nuevo agrupamiento con los grupos que se deseen realizar.

Con la segunda opción habrá que elegir si se desea una introducción manual o aleatoria de los alumnos en los grupos, el número de grupos que se desea crear, cuantos alumnos por grupo y el nombre del agrupamiento a introducir.

- Si la introducción es manual se deberán introducir uno a uno los alumnos mediante su identificador en el sistema.
- Si la introducción es aleatoria se introducirán de manera automática los alumnos elegidos por el profesor (todos los matriculados o sólo los activos en el curso).

- **Modificar equipos:** Una vez asignado un agrupamiento con sus respectivos equipos, si se desea, se pueden modificar los equipos que el agrupamiento posee. Para realizar esto hay dos posibilidades: modificar de manera manual los grupos (borrar alumnos, introducir alumnos, borrar grupos o introducir un nuevo grupo) o sobrescribir todos los grupos mediante la creación de unos nuevos eligiendo sus características con las mismas opciones que la inicialización de alumnos al crear un nuevo agrupamiento.
 - **Tarea calificada:** Una vez realizados los grupos, se podrán calificar y por tanto en esta tabla se mostrará el estado de la calificación (si, parcialmente o no). Este estado es un enlace que nos permite entrar en calificar tarea.
- **Alumno:**

Un alumno tiene la posibilidad de inscribirse en un grupo ya existente.

Un profesor puede permitir a los alumnos poder crear sus propios equipos e incluso poder eliminar miembros si así lo desean.

Un alumno no podrá pertenecer a dos equipos al mismo tiempo en una misma actividad.

Calificación de los equipos

En esta parte del módulo se mostrará una tabla con una lista de las actividades creadas en el curso que son calificables. La información mostrada de la actividad contiene su nombre que será un enlace a la página de la misma y la fecha de entrega (si la posee). Sólo se podrá calificar una tarea si se han creado equipos para la misma, por ello en la tabla pueden aparecer dos enlaces distintos:

- **Inicializar equipos:** este enlace se mostrará cuando la actividad no tiene asociado ningún agrupamiento. Este botón llevará a la misma página que el botón de administración de equipos.
- **Tarea calificada:** si hay grupos, se mostrará el estado de la calificación: si, parcialmente o no.

A la hora de introducir las calificaciones se mostrará una lista de los grupos con sus miembros con la posibilidad de introducir una calificación a cada grupo.

Si la calificación grupal dista de la individual introducida en el libro de calificaciones se le mostrará al profesor un aviso sobre esta situación.

3.2.2. Requisitos técnicos de Moodle

Todo el modulo desarrollado en este proyecto está contenido en *Moodle* y por tanto se necesitan una serie de requisitos propios de *Moodle*.

- Requisitos de *Hardware*[3]: se necesitarán como mínimo:
 - Espacio en disco: 160MB libres.
 - Respaldos: 160MB para copias de seguridad.
 - Procesador: 1GHz.
 - Memoria: 256MB.

- Requisitos de *Software*[4]: los requisitos para la versión de *Moodle* instalada(2.6) son los siguientes:
 - Sistemas operativos *Linux*, *Windows*, *Solaris MacOS X* o *Netware 6*.
 - Un servidor web, primariamente *Apache* o *IIS*.
 - Necesidad de *PHP*: la versión mínima para *Moodle 2.6* es 5.3.3
 - Una base de datos: *MySQL*(versión mínima 5.1.33) y *PostgreSQL*(versión mínima 8.3) son las bases de datos para el desarrollo primario más probadas y con abundante información y soporte.
 - Navegador mínimo para acceder a *Moodle*: versiones recientes de *Firefox*, *Internet Explorer 9*, *Safari 6*, *Google Chrome 11*.

Parte 2. Trabajo desarrollado

Capítulo 4. Teoría

4.1. Introducción

El primer paso para poder realizar la adaptación del módulo propuesto es comprender y conocer el funcionamiento de un módulo de *Moodle*, estudiando los distintos módulos implementados en la plataforma. Por otro lado, se realizó un estudio del módulo *Blended* anteriormente implementado para así poder observar cuales eran sus funciones principales y poder, de esta manera, adaptarlo a la nueva versión de la plataforma.

Tras conocer el funcionamiento de un módulo, se estudió el modelado de los datos dado como tablas en la base de datos. Con las tablas de *Blended* anteriormente implementadas y con el resto de tablas de *Moodle* se comprendió el funcionamiento de las mismas.

Una vez realizado estos estudios, se procedió a la adaptación del código anteriormente introducido en el módulo. Tras esto, se comenzó a programar los distintos ficheros que permiten el uso de los agrupamientos a la hora de administrar y calificar equipos para las distintas actividades que se pueden introducir en *Moodle*, surgiendo la necesidad de la creación de nuevas tablas en el módulo que me permitieran relacionar este módulo con los agrupamientos de *Moodle*.

4.2. Esquema de un módulo

El primer paso realizado ha sido el estudio de la creación de un módulo en *Moodle* para así comprender como se creó el módulo *Blended* y poder utilizarlo de manera correcta.

Cada módulo de almacena en su propia carpeta en el directorio *Moodle/mod*. Todos poseen la misma estructura general [5][6]:

- *Backup/*: directorio que contiene los ficheros necesarios para la realización de copias de seguridad/restauraciones de las actividades de un curso. Como este módulo se ha realizado para la versión *Moodle 2.x*

este directorio está formado por una carpeta con el nombre “*moodle2*” que contendrá los ficheros necesarios para la realización de copias de seguridad/restauraciones en estas versiones de *Moodle*.

- *db/*: directorio que contiene:
 - o *access.php*: fichero donde se definen las capacidades que el *plugin* creará.
 - o *install.xml*: fichero utilizado en la instalación del módulo. Incluye las tablas y campos de la base de datos que el módulo creará.
 - o *upgrade.php*: fichero que maneja la actualización del módulo en la última versión introducida.
- *lang/*: directorio que poseerá una carpeta para cada idioma que necesite ser creado para el módulo. Cada fichero de idioma contendrá todas las cadenas que va a usar el *plugin*.
- *pix/*: fichero donde se almacenarán las imágenes que se desean mostrar en el módulo.
- *index.php*: fichero que es utilizado por *Moodle* cuando se enumeran todas las instancias de un módulo concreto.
- *lib.php*: archivo que contiene las funciones “básicas” que necesita el módulo. Posee dos partes diferenciadas:
 - o Todas las funciones básicas que *Moodle* necesita para que el módulo funcione.
 - o Todas las funciones específicas del módulo, cuyos nombres deben comenzar por el nombre del módulo (en nuestro caso: *blended_*).

Las funciones mínimas que este archivo (para nuestro módulo) debe tener son:

- *Blended_add_instance(\$blended)*: función que se ejecuta cuando un profesor crea una instancia en *Blended*.
- *Blended_update_instance(\$blended)*: función que se ejecuta cuando un profesor actualiza una instancia de *Blended*.
- *Blended_delete_instance(\$id)*: función que es ejecutada cuando un profesor borra una instancia de *Blended*.
- *Blended_user_outline()*: función que proporciona un resumen de la actividad de un usuario.
- *Blended_user_complete(\$course,\$user,\$mod,\$blended)*: función que proporciona un informe detallado de la actividad de un usuario.
- *Blended_print_recent_activity(\$course,\$isteacher,\$timestart)*: función que encuentra la actividad reciente en el módulo.

- *Blended_cron()*: función que se ejecuta de manera periódica buscando tareas a realizar como por ejemplo mandar un *e-mail*.
- *mod_form.php*: archivo utilizado para añadir/editar un módulo de un curso de *Moodle*.
- *versión.php*: fichero que contiene la versión del módulo y otros atributos como por ejemplo la versión de *Moodle* que requiere.
- *view.php*: página de inicio de la actividad.

4.3. Diseño de la base de datos

Toda la información recogida en *Moodle* es guardada en tablas en su base de datos. Cada módulo puede poseer sus propias tablas, que deben seguir una estructura fijada por la plataforma [7]:

- Cada tabla debe tener un campo autonumérico *id* (INT10) como clave primaria.
- La tabla principal que contiene instancias del módulo debe tener el mismo nombre que el módulo y contener al menos los campos:
 - *id*
 - *course*: identificador del curso al que la instancia pertenece
 - *name*: nombre completo de la instancia
- Los nombres de las tablas y de los campos deben evitar el uso de palabras reservadas de la base de datos [8]. Además, los nombres de los campos deberían ser sencillos y cortos.
- La mayoría de las tablas tienen que tener un campo *timestamp* que será actualizado con la fecha actual.

4.3.1. Tablas

En el desarrollo del módulo se han visto utilizadas diferentes tablas:

1. **Tablas de *Blended***: Se han utilizado tablas ya creadas en la versión anterior siendo modificadas para adaptarse a la nueva situación planteada y se ha creado una nueva que relacionara las tablas *Blended* con las de los agrupamientos de *Moodle*.

◆ **Tabla blended:** descripción de las instancias del módulo *Blended*.

Tabla <i>blended</i>		
Nombre	Tipo	Descripción
id	int(10)	Identificador de la instancia del módulo <i>Blended</i> .
course	int(10)	Identificador del curso
name	varchar(255)	Nombre de la instancia del módulo <i>Blended</i> .
intro	longtext	Descripción de la instancia del módulo.
introformat	smallint(4)	Formato del texto del campo de descripción del módulo.
idmethod	bigint(10)	Método de identificación.
idtype	bigint(10)	Tipo de identificador.
codebartype	varchar(5)	Tipo de <i>Codebar</i> . Por defecto: QR2D.
lengthuserinfo	bigint(10)	Longitud del campo del identificador personalizado.
teammethod	bigint(10)	Método de creación de los equipos
numteams	bigint(10)	Número de equipos por defecto.
nummembers	bigint(10)	Número máximo de estudiantes por equipo.
assignment	bigint(10)	La tarea para la que se crearán los equipos.
randomkey	bigint(20)	Clave aleatoria para codificar el identificador del estudiante mediante operación XOR.

Tabla 1: Tabla *blended*

◆ **Tabla *blended_team*:** cada entrada en la tabla representa un equipo creado para cada tarea en la que se ha realizado la asignación de equipos. El campo *id_team* representa el identificador del equipo creado en las tablas de agrupamientos. Cuando el grupo es creado por un alumno el campo *userid_leader* contendrá el identificador del alumno que ha creado el equipo, si por el contrario es creado por un profesor su valor será cero.

Tabla <i>blended_team</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de cada equipo en <i>Blended</i>
id_team	int(4)	Identificador de cada equipo de los agrupamientos
id_assignment	bigint(10)	Identificador de la tarea en que se encuentra el equipo.
name_team	varchar(255)	Nombre del equipo
userid_leader	bigint(10)	Entero formado por el identificador del alumno creador del equipo

Tabla 2: Tabla *blended_team*

- ◆ **Tabla *blended_member***: En esta tabla son introducidos los miembros de los equipos definidos en la tabla *blended_team*.

Tabla <i>blended_member</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de cada estudiante para cada tarea en la tabla
userid	bigint(13)	El <i>id</i> de la tabla “user” de cada miembro
id_member	varchar(13)	El identificador correspondiente a su DNI sin la letra final
id_team	bigint(10)	El <i>id</i> del equipo al que pertenece el estudiante. Se corresponde con el valor del campo <i>id_team</i> de la tabla <i>blended_team</i> .
leader	tinyint(1)	Booleano que indica quien ha creado el equipo (0 si el profesor, 1 si el alumno)

Tabla 3: Tabla *blended_member*

- ◆ **Tabla *blended_grade***: cada entrada en la tabla representa un equipo que ha sido calificado por un profesor para una tarea específica.

Tabla <i>blended_grade</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la entrada en la tabla
id_assignment	bigint(10)	Identificador de la actividad que está siendo calificada
id_assignment_0	bigint(10)	Identificador de la actividad en la que fue creado el equipo
id_team	bigint(10)	El <i>id</i> del equipo que está siendo calificado
grade	bigint(10)	Calificación introducida para el equipo
rewrite	tinyint(1)	Booleano que indica si la nota se puede sobrescribir (1) o no (0).

Tabla 4: Tabla *blended_grade*

- ◆ **Tabla *blended_assign_grouping***: cada registro de la tabla relaciona a cada actividad con el agrupamiento que se le ha asignado a la hora de crear los grupos.

Tabla <i>blended_assign_grouping</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la entrada en la tabla
id_assign	bigint(10)	Identificador de la actividad
id_grouping	bigint(10)	Identificador del agrupamiento que se le ha asignado a esa actividad.

Tabla 5: Tabla *blended_assign_grouping*

2. **Tablas de *groupings*:** A la hora de la realización de la adaptación de los agrupamientos de *Moodle* al módulo *Blended*, ha sido necesario el uso de las tablas asociadas a estos agrupamientos.

◆ **Tabla *groupings*:** Cada registro de la tabla se corresponde con un agrupamiento creado en *Moodle*.

Tabla <i>groupings</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la instancia
courseid	bigint(10)	Identificador del curso en que ha sido creada la instancia
name	varchar(255)	Nombre del agrupamiento
idnumber	varchar(100)	Identificador arbitrario dado por el módulo responsable
description	longtext	Descripción de la instancia del módulo
descriptionformat	tinyint(2)	Formato del texto del campo <i>description</i>
configdata	longtext	Datos para configuraciones extra
timecreated	bigint(10)	Marca de tiempo en el que el agrupamiento ha sido creado
timemodified	bigint(10)	Última marca de tiempo en el que el agrupamiento ha sido modificado.

Tabla 6: Tabla *groupings*

◆ **Tabla *groupings_groups*:** En esta tabla se almacenan los grupos que posee cada agrupamiento.

Tabla <i>groupings_groups</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la instancia
groupingid	bigint(10)	Identificador del agrupamiento
groupid	bigint(10)	Identificador del grupo
timeadded	bigint(10)	Marca de tiempo en el que el grupo ha sido creado

Tabla 7: Tabla *groupings_groups*

- ◆ **Tabla *groups***: En esta tabla se almacenan todos los grupos que han sido creados. El identificador de esta tabla se hace corresponder con el campo *id_team* de la tabla *blended_team* para que las tablas del módulo *blended* se encuentren sincronizadas con las de los agrupamientos.

Tabla <i>groups</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la instancia
courseid	bigint(10)	Identificador del curso en que ha sido creada la instancia
name	varchar(100)	Nombre del grupo
idnumber	varchar(100)	Identificador arbitrario dado por el módulo responsable.
description	longtext	Descripción de la instancia del módulo
descriptionformat	tinyint(2)	Formato del texto del campo <i>description</i>
enrolmentkey	varchar(50)	Contraseña para inscribirse en un grupo. Por defecto posee el valor <i>null</i>
picture	bigint(10)	Imagen que identifica al grupo
hidepicture	tinyint(1)	Booleano que indica si la imagen es o no mostrada.
timecreated	bigint(10)	Marca de tiempo en el que el agrupamiento ha sido creado
timemodified	bigint(10)	Última marca de tiempo en el que el agrupamiento ha sido modificado.

Tabla 8: Tabla *groups*

- ◆ **Tabla *groups_members***: Cada entrada en la tabla se corresponde con los miembros que pertenecen a los equipos creados.

Tabla <i>groups_members</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la instancia
groupid	bigint(10)	Identificador del grupo
userid	bigint(10)	Identificador del usuario
timeadded	bigint(10)	Marca de tiempo en el que el miembro ha sido añadido
component	varchar(100)	Define el componente de <i>Moodle</i> que añadió la pertenencia del usuario al grupo (e.g. ' <i>auth_myplugin</i> ') o en blanco si se agregó manualmente.
itemid	bigint(10)	Si el campo <i>component</i> se establece, <i>itemid</i> puede ser utilizado para definir la instancia del componente que creó la entrada. Si <i>component</i> está vacío el valor de este campo es por defecto cero.

Tabla 9: Tabla *groups_members*

3. **Tablas de *grades*:** Cuando se introducen calificaciones, se actualiza el libro de calificaciones. Por tanto, y para poder observar si hay diferentes calificaciones en el libro y en la tabla *blended_grade* y en ese caso avisar al profesor de la situación, se han utilizado las siguientes tablas:

- ◆ **Tabla *grades_items*:** Esta tabla mantiene información sobre los elementos que son calificables. Esta información será utilizada en el módulo *Blended* ya que sólo aquellas actividades que sean calificables se les podrán asignar agrupamientos.

Tabla <i>grade_items</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la instancia.
courseid	bigint(10)	Identificador del curso
categoryid	bigint(10)	Categoría grupal a la que pertenece la instancia (opcional)
itemname	varchar(255)	Nombre de la actividad
itemtype	varchar(30)	Tipo de instancia ('mod', 'blocks', 'import', etc)
itemmodule	varchar(30)	Nombre de la instancia a la que pertenece la actividad ('forum', 'quiz', etc)
iteminstance	bigint(10)	Id del módulo de la actividad
itemnumber	bigint(10)	Puede ser usado para distinguir múltiples calificaciones para una actividad
iteminfo	bigint(10)	Información y notas sobre la actividad
idnumber	varchar(255)	Identificador arbitrario dado por el módulo responsable

calculation	longtext	Fórmula que describe como derivar esta calificación desde otras actividades
gradetype	smallint(4)	Tipo de calificación: 0=ninguna, 1=valor, 2=escala, 3=texto
grademax	decimal(10,5)	Calificación máxima que se puede otorgar
grademin	decimal(10,5)	Calificación mínima que se puede otorgar
scaleid	bigint(10)	Si la calificación está basada en una escala, identifica qué escala es
outcomeid	bigint(10)	Si la calificación se relaciona con un resultado, identifica cual es.
gradepass	decimal(10,5)	Indica que calificación se va a pasar
multifactor	decimal(10,5)	Factor por el que van a multiplicar todas las calificaciones
plusfactor	decimal(10,5)	Factor que se le va a sumar a todas las calificaciones
aggregationcoef	decimal(10,5)	Coefficiente de agregación utilizado para la categoría de pesos u otros tipos de agregación
sortorder	bigint(10)	Tipo de orden que poseen las columnas
display	bigint(10)	Indica cómo se va a mostrar: calificaciones reales, porcentajes, letras o el valor del curso por defecto (0)
decimals	tinyint(1)	Precisión, número de dígitos decimales que se van a mostrar
hidden	bigint(10)	Si posee el valor 1, la calificación se ocultará
locked	bigint(10)	Si posee el valor 1, la calificación no podrá ser modificada
locktime	bigint(10)	Marca de tiempo a partir de la cual se bloquearán todas las calificaciones finales
needsupdate	bigint(10)	Si se establece como indicador, a continuación, se volverá a calcular toda la columna
timecreated	bigint(10)	Marca de tiempo en el que la calificación fue creada
timemodified	bigint(10)	Ultima marca de tiempo en el que la calificación fue modificada

Tabla 10: Tabla *grade_items*

- ◆ **Tabla *grades_grades*:** Esta tabla contiene las calificaciones individuales para cada usuario y cada actividad. Estas calificaciones serán las mostradas en el libro de calificaciones del curso.

Tabla <i>grade_grades</i>		
Nombre	Tipo	Descripción
id	bigint(10)	Identificador de la instancia
itemid	bigint(10)	Identificador de la actividad a la que pertenece la calificación
userid	bigint(10)	Identificador del usuario que está siendo calificado
rawgrade	decimal(10,5)	Indica si la calificación posee formato decimal
rawgrademax	decimal(10,5)	Máxima calificación permitida, establecida cuando fue creada la instancia
rawgrademin	decimal(10,5)	Mínima calificación permitida, establecida cuando fue

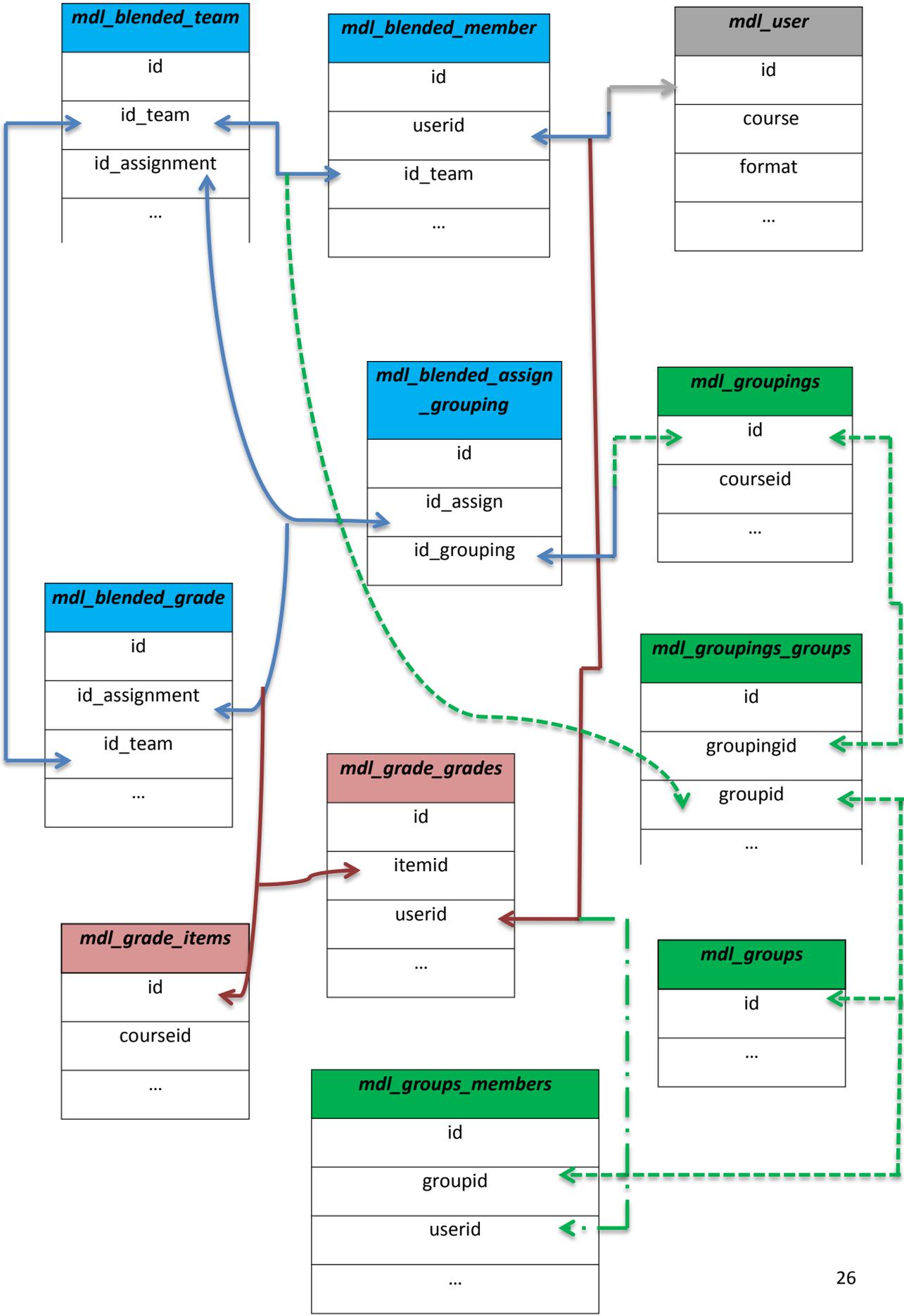
		creada la instancia
rawscaleid	bigint(10)	Si la calificación está basada en una escala, indica que escala es
usermodified	bigint(10)	Identificador de la última persona que ha modificado la calificación
finalgrade	decimal(10,5)	Calificación final
hidden	bigint(10)	Si posee el valor 1, la calificación se ocultará
locked	bigint(10)	Si posee el valor 1, la calificación no podrá ser modificada
locktime	bigint(10)	Marca de tiempo a partir de la cual se bloquearán todas las calificaciones finales
exported	bigint(10)	Fecha en la que la última calificación fue exportada. Por defecto posee el valor cero
overridden	bigint(10)	Campo que indica si la calificación ha sido sobrescrita en el libro de calificaciones. Si se ha sobrescrito, en este campo se introduce la marca de tiempo en la que la modificación fue realizada
excluded	bigint(10)	Indica si la calificación fue excluida de las funciones de agregación.
feedback	longtext	Retroalimentación de las calificaciones
feedbackformat	bigint(10)	Formato del texto del campo <i>feedback</i> .
information	longtext	Información adicional
informationformat	bigint(10)	Formato del texto del campo <i>information</i>
timecreated	bigint(10)	Marca de tiempo en el que la calificación fue creada
timemodified	bigint(10)	Ultima marca de tiempo en el que la calificación fue modificada

Tabla 11: Tabla *grade_grades*

4.3.2. Diagrama Entidad - Relación

En esta sección se va a mostrar un diagrama entidad-relación que me relacione las tablas que han sido utilizadas a la hora de adaptar el módulo, además de las básicas utilizadas para que las tablas descritas con anterioridad puedan funcionar. De esta manera se podrá observar como los campos de las distintas tablas se encuentran interrelacionados.

Tabla 12: Diagrama entidad-relación



4.4. Ficheros del módulo

En esta sección se van a explicar todos los ficheros que han sido necesarios para la adaptación del módulo y la implementación de los agrupamientos en *Moodle 2.x*. El nuevo módulo *Blended* posee la misma estructura que la versión antigua introduciendo los cambios necesarios para poder desarrollar la actividad propuesta. Todos los ficheros se encontrarán en */mod/blended*.

Todos los ficheros que se han desarrollado están programados en lenguaje *PHP* (*Hipertext Preprocesor*) [9]. *PHP* es un lenguaje de código abierto muy popular, especialmente adecuado para el desarrollo web y que puede ser incrustado en *HTML*.

Lo que distingue a *PHP* de otros tipos de lenguaje como por ejemplo *Javascript* del lado del cliente es que el código es ejecutado en el servidor, generando *HTML* y enviándolo al cliente que recibirá el resultado de ejecutar el script. El servidor web puede ser incluso configurado para que procese todos los ficheros *HTML* con *PHP*.

Lo mejor de usar *PHP* es que es extremadamente simple para el principiante, pero a su vez ofrece muchas características avanzadas para los programadores profesionales.

4.4.1. Ficheros generales del módulo

Como se ha mencionado en la sección anterior, existen ciertos ficheros presentes en todos los módulos necesarios para el correcto funcionamiento de los mismos.

Fichero access.php

Este fichero se encuentra en la carpeta */mod/blended/db* y está encargado de definir qué capacidad creará el *plugin*. Cuando se agregan nuevas capacidades es importante modificar el fichero *version.php* para que los cambios sean efectivos.

Está formado por una serie de permisos que serán utilizados en los distintos ficheros del módulo. A partir de la versión de *Moodle 2.3* se añadió una nueva capacidad llamada *addinstance*. *Moodle* la buscará y si no está creada en un módulo mostrará un mensaje de error de que esa capacidad falta. Por tanto este fichero debe

contener primero esta entrada y tras ella el resto de permisos del módulo. Para ver la estructura del fichero, se mostrará la primera entrada de éste:

```
$capabilities = array (
    'mod/blended:addinstance' => array (
        'riskbitmask' => RISK_XSS,
        'capytype' => 'write',
        'contextlevel' => CONTEXT_COURSE,
        'archetypes' => array (
            'editingteacher' => CAP_ALLOW,
            'manager' => CAP_ALLOW
        ),
        'clonepermissionsfrom' => 'moodle/course:manageactivities'
    ),
    'mod/blended:blended' => array (
        'capytype' => 'read',
        'contextlevel' => CONTEXT_MODULE,
        'legacy' => array (
            'guest' => CAP_PREVENT,
            'student' => CAP_ALLOW,
            'teacher' => CAP_ALLOW,
            'editingteacher' => CAP_ALLOW,
            'manager' => CAP_ALLOW
        )
    ),
)
```

- */mod/blended:addinstance*: nombre utilizado para este permiso
- *riskbitmask*: debido a la capacidad de un usuario de realizar distintos tipos de tareas se introduce cierto riesgo en la seguridad. Por tanto, cada capacidad debe enumerar cuales son los riesgos. En este caso se ha utilizado el valor *RISK_XSS* (un usuario puede enviar contenido que no ha sido “limpiado”; originalmente está protegido por *isteacher()*).
- *capytype*: puede poseer dos valores: *read*(sólo lectura) o *write*(permite realizar cambios)
- *contextlevel*: es el permiso más importante. Todas las capacidades del módulo deben tener el valor *CONTEXT_MODULE*.
- *archetypes*: define para cada tipo de rol que permisos deberían tener esos roles cuando se instala el módulo. Los valores que pueden poseer son: *CAP_ALLOW* (permiso concedido), *CAP_PREVENT* (permiso denegado) y *CAP_PROHIBIT* (permiso prohibido).

- *legacy*: indica los permisos que posee cada rol dentro de esa capacidad. Pueden poseer los mismos valores que *archetypes*.

El resto de entradas de este fichero se mencionarán en la descripción de cada uno de ellos.

Fichero upgrade.php

Este archivo está encargado de actualizar el módulo para que coincida con la última versión definida en el fichero *version.php*. Para poder realizar una correcta actualización será necesario:

- Añadir las nuevas columnas al fichero *install.xml*
- Añadir las instrucciones a *upgrade.php*
- Actualizar la versión en *version.php*

En este proyecto se ha tenido que introducir una nueva tabla y una nueva columna a una tabla existente en la base de datos y por tanto y tras actualizar el fichero *install.xml*, se han realizado los siguientes cambios en *upgrade.php*:

```

if ($result && $oldversion < 2014071406) {
    $table = new xmldb_table('blended_assign_grouping');
    $table->add_field('id',XMLDB_TYPE_INTEGER, '10',null, XMLDB_NOTNULL,
        XMLDB_SEQUENCE,null);
    $table->add_field('id_assign',XMLDB_TYPE_INTEGER,'10',null,
        XMLDB_NOTNULL,null,null,'id');
    $table->add_field('id_grouping',XMLDB_TYPE_INTEGER,'10',null,
        XMLDB_NOTNULL,null,null,'id_assign');
    $table->add_key('primary', XMLDB_KEY_PRIMARY, array('id'));
    if(!$dbman->table_exists($table)){
        $dbman->create_table($table);
    }

    $table1=new xmldb_table('blended_team');
    $field=new xmldb_field('id_team',XMLDB_TYPE_INTEGER,'4',null,
        XMLDB_NOTNULL,null,'0','id');
    if(!$dbman->field_exists($table1, $field)){
        $dbman->add_field($table1, $field);
    }

    upgrade_mod_savepoint(true, '2014071406', 'blended');
}

```

En este caso, si la versión que posee *Moodle* de *Blended* es inferior a “2014071406” se añadirán a la base de datos la tabla *blended_assign_grouping* y el campo *id_team* en la tabla *blended_team*.

Fichero index.php

Página utilizada por *Moodle* al enumerar todas las instancias del módulo *Blended* que se encuentra en un curso concreto. El comienzo del fichero debe ser el siguiente:

```
require_once("../../config.php");
require_once("lib.php");

$id = required_param('id', PARAM_INT); // course
global $DB;
if (! $course = $DB->get_record("course", "id", $id))
{
    error("Course ID is incorrect");
}
```

Tras esto se definirán el resto de operaciones que el módulo requiera implementar.

Fichero mod_form.php

Este fichero es utilizado a la hora de añadir/modificar un módulo para un curso. Contiene los elementos que se mostrarán en el formulario responsable de crear/installar una instancia en el módulo. La clase deberá ser llamada para nuestro módulo: *mod_blended_mod_form*.

Primero se crea un elemento de texto llamado “name” que contendrá el nombre de la instancia, además de una cabecera implementada por el método *addElement*. Mediante la función *select* se inicializarán distintos valores de las listas desplegadas.

Al final del fichero se introducen las funciones *standard_coursemodule_elements* que añade elementos comunes a todos los módulos y *add_action_buttons* que añade los botones de enviar o cancelar del formulario. Los datos son pasados a *blended_update_instance* o *blended_add_instance* dependiendo de si se entra añadiendo o actualizando una instancia.

Fichero version.php

Archivo esencial que se ubica en el directorio raíz */mod/blended*. Contiene una serie de propiedades que se utilizan durante la instalación del módulo y el proceso de actualización. Permite asegurarse que el módulo es compatible con *Moodle* así como la detección de que es necesaria una actualización. En este módulo este fichero contendrá:

```
defined('MOODLE_INTERNAL') || die;

$module->version = 2014071406;
    $module->requires = 2013051407.00;
$module->component = 'mod_blended';
$module->cron      = 0;
```

- *\$module->version*: define el número de versión de *Blended*. El formato es de la forma: “AAAAMMDDXX” con “AAAA” el año, “MM” el mes, “DD” el día y “XX” un contador incremental.
- *\$module->requieres*: especifica el número mínimo de versión de *Moodle* que *Blended* requiere. Posee la misma estructura que el elemento anterior. En este caso el valor introducido es el de la versión de *Moodle 2.5* para que sea compatible con las versiones de *Moodle* iguales o superiores a la indicada.
- *\$module->component*: declara el tipo y el nombre del *plugin*.
- *\$module->cron*: permite el acelerador de llamadas de función *cron*. En este caso se ha establecido un cero ya que la función *cron* existe.

Fichero view.php

Es el fichero que muestra la página principal de una instancia del módulo *Blended*. En esta página se encuentran los distintos enlaces al conjunto de funciones que este módulo posee. Este fichero posee los siguientes permisos definidos en *access.php*:

```
'mod/blended:view' => array(
    'captive' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'archetypes' => array(
        'student' => CAP_ALLOW,
        'frontpage' => CAP_ALLOW,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

De esta manera sólo los roles especificados pueden entrar en este fichero y por tanto acceder a la página principal de *Blended*.

Antes de mostrar la lista de *links* a las distintas actividades dentro del módulo se realiza una comprobación de que hay estudiantes matriculados en el curso. Si no hay ningún estudiante matriculado los *links* aparecerán desactivados.

Por otro lado, se mostrarán distintos enlaces dependiendo del tipo de rol que posea el usuario. Para ello se hace uso de la capacidad *rolelinks* definida en *access.php*

```
'mod/blended:rolelinks' => array (
    'capytype' => 'read',
    'contextlevel' => CONTEXT_MODULE
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_PREVENT,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

- Si el rol se corresponde con un profesor poseerá los enlaces
 - “Administración de equipos”
 - “Calificar equipos”
- Si el rol se corresponde con un alumno tendrá el enlace “inscribirse en equipo”.

En ambos roles aparecerán los enlaces a “generar página de tarea” e “imprimir página de etiquetas”

Por último, si el rol se corresponde con *teacher* se podrán mostrar otros *links* que informen al profesor si hay estudiantes no activos, que no posean código de identificación o que simplemente no haya ningún estudiante activo o matriculado. Para estos avisos se hace uso de la capacidad *viewalerts*:

```

'mod/blended:viewalerts' => array (

    'captype' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_PREVENT,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),

```

Fichero lib.php

Fichero que contiene la lista de funciones que serán utilizadas en el módulo.

1. Funciones esenciales que todo módulo de *Moodle* debe poseer: Entre ellas caben destacar:

function blended_add_instance (\$blended)	
Descripción	Función encargada de crear una nueva instancia.
Parámetros	<i>\$blended</i> Objeto de la instancia del módulo blended.
Return	Devuelve el identificador de la nueva instancia.

Tabla 13: Función *blended_add_instance*

function blended_update_instance (\$blended)	
Descripción	Dado un objeto que contenga todos los datos necesarios, esta función actualiza una instancia ya existente con los nuevos datos.
Parámetros	<i>\$blended</i> Objeto de la instancia del módulo blended.
Return	Devuelve un booleano que indica si la actualización ha tenido éxito o no.

Tabla 14: Función *blended_update_instance*

function blended_delete_instance (\$id)	
Descripción	Dado un identificador de una instancia del módulo, esta función borrará de manera permanente la instancia y cualquier dato que dependa de ella
Parámetros	<i>\$id</i> Identificador de la instancia del módulo
Return	Devuelve un booleano que indica si la actualización ha tenido éxito o no

Tabla 15: Función *blended_delete_instance*

2. Funciones propias del módulo *Blended*: en esta sección se van a presentar todas las funciones que han sido utilizadas a la hora del desarrollo de las actividades del módulo.

function blended_gen_cleanidvalue (\$user, \$blended)					
Descripción	<p>Genera el código numérico del identificador del estudiante a partir de los parámetros especificados en la instancia del módulo.</p> <p>No aplica filtrado ni codificación. Selecciona el tipo de identificación:</p> <ul style="list-style-type: none"> - user->idnumber - user_info_data->data <p>Asegurándose de quitar el guión y la letra.</p> <p>Ver: <i>blended_clean_idnumber()</i></p>				
Argumentos	<table border="1"> <tr> <td><i>\$user</i></td> <td>Objeto de la tabla "user" perteneciente al estudiante.</td> </tr> <tr> <td><i>\$blended</i></td> <td>Objeto de la instancia del módulo blended.</td> </tr> </table>	<i>\$user</i>	Objeto de la tabla "user" perteneciente al estudiante.	<i>\$blended</i>	Objeto de la instancia del módulo blended.
<i>\$user</i>	Objeto de la tabla "user" perteneciente al estudiante.				
<i>\$blended</i>	Objeto de la instancia del módulo blended.				
Return	Devuelve el código del identificador del estudiante o -1 si <i>user->idnumber</i> no ha sido introducido aún				

Tabla 16: Función *blended_gen_cleanidvalue*

function blended_clean_idnumber (\$ean)	
Descripción	Elimina del <i>user->idnumber</i> (DNI) el guión y la letra ya que en este campo sólo deben figurar dígitos.
Argumentos	<i>\$ean</i> Cadena <i>\$user->idnumber</i>
Return	Devuelve la cadena <i>user->idnumber</i> sin guión ni letra.

Tabla 17: Función *blended_clean_idnumber*

function blended_update_grouping (\$courseid,\$steamname,\$mem,\$groupingid,\$ya,\$groupid=null)		
Descripción	<p>Actualización de las tablas de agrupamientos debido a cambios realizados en sus grupos. Se pueden realizar dos tareas:</p> <ul style="list-style-type: none"> - Si existe el grupo en el agrupamiento dado se actualizará con los nuevos miembros introducidos mediante <i>\$mem</i>. - Si no existe el grupo, se introducirá uno nuevo con los miembros <i>\$mem</i>. <p>Ver:</p> <ul style="list-style-type: none"> - <i>groups_get_group_by_name()</i> - <i>blended_update_group()</i> - <i>blended_add_new_group()</i> 	
Argumentos	<i>\$courseid</i>	Identificador del objeto de la instancia del curso
	<i>\$steamname</i>	Cadena que posee el nombre del equipo
	<i>\$mem</i>	Array con los miembros que posee el equipo
	<i>\$groupingid</i>	Identificador del agrupamiento al que pertenece el grupo
	<i>\$ya</i>	Parámetro pasado por defecto a las otras funciones cuyo valor será explicado en ellas
	<i>\$groupid</i>	Identificador del grupo
Return	Si la opción elegida es crear un nuevo grupo, se devolverá el valor del identificador del nuevo grupo creado	

Tabla 18: Función *blended_update_grouping*

function blended_update_group (\$courseid,\$steamname,\$mem,\$groupid,\$ya)		
Descripción	<p>Actualización de un grupo de un agrupamiento. Esta actualización puede ser llevada a cabo por un profesor o por un alumno.</p> <ul style="list-style-type: none"> - Si <i>\$ya=null</i> indica que la acción es llevada a cabo por un profesor - Si <i>\$ya!=null</i> la función es llamada por un alumno <p>Ver:</p> <ul style="list-style-type: none"> - <i>groups_remove_member()</i> - <i>groups_update_group()</i> - <i>blended_clean_idnumber()</i> - <i>groups_add_member()</i> - <i>groups_update_grouping()</i> 	
Argumentos	<i>\$courseid</i>	Identificador del objeto de la instancia del curso.
	<i>\$steamname</i>	Cadena que posee el nombre del equipo
	<i>\$mem</i>	Array con los miembros que posee el equipo
	<i>\$groupid</i>	Identificador del grupo que se está actualizando
	<i>\$ya</i>	Parámetro que diferencia el tipo de usuario que llama a la función

Tabla 19: Función *blended_update_group*

function blended_add_new_group (\$courseid,\$steamname,\$mem,\$groupingid)		
Descripción	<p>Inserta un nuevo grupo en el agrupamiento cuyo identificador viene dado por <i>\$groupingid</i>. Recibe los <i>arrays</i> <i>\$mem</i> y <i>\$steamname</i> con los datos a introducir. Tras realizar una comprobación del número de miembros que posee cada grupo, se hace una diferenciación ya que</p> <ul style="list-style-type: none"> - Si es mayor que uno <i>\$mem</i> es un array - Si es 1 puede ser: <ul style="list-style-type: none"> • Un array si la orden procede de un profesor • Un valor si posee de un alumno. <p>Se busca la existencia de los miembros en la base de datos ya que los miembros introducidos que no existan no se almacenan. Por último, se asigna el grupo al agrupamiento. Ver:</p> <ul style="list-style-type: none"> - <i>groups_create_group()</i> - <i>blended_clean_idnumber()</i> - <i>groups_add_member()</i> - <i>groups_assign_grouping()</i> 	
Argumentos	<i>\$courseid</i>	Identificador del objeto de la instancia del curso
	<i>\$steamname</i>	Cadena que posee el nombre del equipo
	<i>\$mem</i>	Array con los miembros que posee el equipo
	<i>\$groupingid</i>	Identificador del agrupamiento en el que va a ser almacenado el grupo
Return	Devuelve el identificador del grupo creado	

Tabla 20: Función *blended_add_new_group*

function blended_method_radom (\$studentsselection,\$course,\$id_assignment=null)		
Descripción	<p>Generación del método <i>random</i> en la creación de equipos. Según el tipo de alumnos escogido pasado en el parámetro <i>\$studentsselection</i> se recogerán mediante la función <i>blended_get_course_students_ids()</i> los identificadores de los alumnos bien sean sólo los activos o todos los matriculados.</p> <p>Ver <i>blended_get_course_students_ids()</i></p>	
Argumentos	<i>\$studentsselection</i>	Cadena que contiene el tipo de alumnos que se ha escogido (activos o todos los matriculados)
	<i>\$course</i>	Objeto de la instancia del curso
	<i>\$id_assignment</i>	Identificador de la tarea en la que se están generando los equipos.
Return	<p>Devuelve un <i>array</i> formado por:</p> <ul style="list-style-type: none"> - <i>\$freemembers</i>: identificadores de los estudiantes del curso - <i>\$free_index</i>: índice de referencia - <i>\$numfreemembers</i>: número de alumnos recogidos en <i>\$freemembers</i> - <i>\$userids</i>: campo <i>id</i> de los alumnos recogidos en <i>\$freemembers</i> 	

Tabla 21: Función *blended_method_random*

function blended_get_course_students_ids (\$course,\$sid_assignment=null, \$selectactivestudents=false)		
Descripción	<p>Obtiene los estudiantes matriculados en el curso, todos o solamente los activos. Si recibe el identificador de una asignatura sólo devuelve aquellos que no estén inscritos en ningún equipo. Dos modos de funcionamiento:</p> <ol style="list-style-type: none"> 1. Si recibe <i>\$sid_assignment</i> devuelve el array <i>\$freemembers</i> cuyos elementos son los estudiantes matriculados en el curso (activos o todos) ordenados aleatoriamente que aún no han sido asignados a ningún equipo en la tarea con identificador <i>\$sid_assignment</i>. 2. Si no recibe <i>\$sid_assignment</i> obtiene el array <i>\$students</i> cuyos elementos son los estudiantes matriculados en el curso (activos o todos). <p>Dependiendo del valor de <i>\$selectactivestudents</i> (true o false) se seleccionan sólo los estudiantes activos o bien todos matriculados en el curso. Por defecto se seleccionan todos.</p>	
Argumentos	<i>\$course</i>	Objeto de la instancia del curso
	<i>\$sid_assignment</i>	Identificador de la tarea en la que se están generando los equipos
	<i>\$selectactivestudents</i>	Booleano que determina si la selección de alumnos es todos (false) o sólo los activos (true)
Return	Devuelve un <i>array</i> con los estudiantes seleccionados o <i>false</i> si hubo algún error.	

Tabla 22: Función *blended_get_course_students_ids*

function blended_delete_groupings (\$activitycm_or_id)		
Descripción	<p>Borrado del agrupamiento para la tarea elegida. Esta función está encargada de borrar los registros de las tablas de los agrupamientos.</p> <ul style="list-style-type: none"> - Si <i>\$activitycm_or_id</i> es un objeto entonces la actividad para la cual se van a borrar los registros tendrá las características de ese objeto. - Si es un identificador, se obtendrá el objeto actividad mediante la función <i>get_coursemodule_from_id()</i>. <p>Ver:</p> <ul style="list-style-type: none"> - <i>get_coursemodule_from_id()</i> - <i>groups_get_all_groups()</i> - <i>groups_delete_group()</i> 	
Argumentos	<i>\$activitycm_or_id</i>	Objeto/identificador mediante el cual se obtendrá el objeto de la actividad.

Tabla 23: Función *blended_delete_groupings*

function blended_actualizar_agrupamiento (\$groupingid,\$memberid, \$numteams,\$nummembers,\$teamname,\$id_assignment,\$id_team)															
Descripción	<p>Actualiza los grupos y miembros del agrupamiento de la tarea elegida con identificador <i>\$id_assignment</i>.</p> <p>Primero se recogen todos los usuarios registrados en el curso en el <i>array</i> 'lista_usuarios'. Posteriormente se introducen en el <i>array</i> 'lista_idnumbers' los identificadores de los usuarios. Este <i>array</i> se usará para comprobar la existencia del usuario introducido en el sistema</p> <p>Para cada grupo, se comprueba su existencia en la base de datos, si el grupo existe se actualiza y si no existe se creará un nuevo grupo. Una vez comprobado si los usuarios existen, se comprueba si hay algún identificador repetido: para cada miembro del vector 'miembros_grupos' se introduce en el <i>array</i> 'contador' el número de veces que aparece. Tras comprobar los grupos se comprueban los miembros:</p> <ul style="list-style-type: none"> - Se comprueba la existencia de cada miembro introducido en la base de datos ya que si no existe en la base de datos el miembro no se introducirá en el grupo. - Si un grupo está vacío o todos sus miembros no existen en la base de datos se borrará. <p>Tras esto se comprueba si hay grupos del agrupamiento que están calificados para poder actualizar las calificaciones de los miembros de los equipos. Por último se introduce en un <i>array</i> el número de veces que se repite cada identificador.</p> <p>Ver:</p> <ul style="list-style-type: none"> - <i>blended_clean_idnumber()</i> - <i>groups_group_exists()</i> - <i>groups_update_group()</i> - <i>groups_create_group()</i> - <i>groups_delete_group()</i> - <i>groups_add_member()</i> - <i>groups_get_members()</i> 														
Argumentos	<table border="1"> <tbody> <tr> <td><i>\$groupingid</i></td> <td>Identificador del agrupamiento</td> </tr> <tr> <td><i>\$id_assignment</i></td> <td>Identificador de la tarea</td> </tr> <tr> <td><i>\$memberid</i></td> <td>Array con los miembros introducidos en los equipos</td> </tr> <tr> <td><i>\$numteams</i></td> <td>Número de equipos</td> </tr> <tr> <td><i>\$nummembers</i></td> <td>Número de miembros por equipo</td> </tr> <tr> <td><i>\$teamname</i></td> <td>Array con los nombres de los equipos</td> </tr> <tr> <td><i>\$id_team</i></td> <td>Array con los identificadores de cada equipo</td> </tr> </tbody> </table>	<i>\$groupingid</i>	Identificador del agrupamiento	<i>\$id_assignment</i>	Identificador de la tarea	<i>\$memberid</i>	Array con los miembros introducidos en los equipos	<i>\$numteams</i>	Número de equipos	<i>\$nummembers</i>	Número de miembros por equipo	<i>\$teamname</i>	Array con los nombres de los equipos	<i>\$id_team</i>	Array con los identificadores de cada equipo
<i>\$groupingid</i>	Identificador del agrupamiento														
<i>\$id_assignment</i>	Identificador de la tarea														
<i>\$memberid</i>	Array con los miembros introducidos en los equipos														
<i>\$numteams</i>	Número de equipos														
<i>\$nummembers</i>	Número de miembros por equipo														
<i>\$teamname</i>	Array con los nombres de los equipos														
<i>\$id_team</i>	Array con los identificadores de cada equipo														
Return	Devuelve un <i>array</i> con el número de veces que se repite cada identificador en los equipos														

Tabla 24: Función *blended_actualizar_agrupamiento*

function blended_probe_grouping_exist (\$id_assignment)	
Descripción	Comprobación de la existencia de la asignación de un agrupamiento para una actividad con identificador <i>\$id_assignment</i>
Argumentos	<i>\$id_assignment</i> Identificador de la actividad
Return	Devuelve <i>true</i> si existe y <i>false</i> si no existe.

Tabla 25: Función *blended_probe_group_exist*

function blended_update_teams_by_groups (<i>\$id_assignment</i> , <i>\$leader</i> =null, <i>\$group_id</i> =null, <i>\$currentuserid</i> =null)	
Descripción	<p>Actualización de las tablas "blended_team" y "blended_member" con los datos de los agrupamientos.</p> <p>Recibe el identificador de la actividad <i>\$id_assignment</i> y los datos <i>\$leader</i> y <i>\$group_id</i> usados estos últimos cuando el usuario que llama a esta función es un alumno.</p> <p>Primero se recoge el identificador del agrupamiento mediante <i>\$id_assignment</i> de la tabla "blended_assign_grouping". Después se borran de "blended_team" y "blended_member" los registros asociados a esta tarea y ese agrupamiento. Por último se introducen en estas dos tablas los datos de los grupos y miembros del agrupamiento que tiene asociado esa tarea.</p> <p>Si la función es llamada por un alumno, se actualizan las tablas introduciendo en los campos 'userid_leader' y 'leader' de "blended_team" y "blended_member" el valor del identificador del usuario actual y 1 respectivamente y siempre y cuando el alumno tenga la capacidad de líder del grupo.</p> <p>Ver <i>blended_clean_idnumber()</i></p>
Argumentos	<i>\$id_assignment</i> Identificador de la actividad
	<i>\$leader</i> Constante que define si la petición es realizada por un alumno
	<i>\$group_id</i> Identificador del grupo proporcionado cuando <i>\$leader</i> != null
	<i>\$currentuserid</i> Identificador del usuario actual

Tabla 26: Función *blended_update_teams_by_groups*

function blended_num_members (\$members_group,\$action=null)	
Descripción	<p>Comprueba la existencia de los usuarios introducidos y repetición de los identificadores de los mismos en distintos grupos.</p> <p>Recibe el array <i>\$members_group</i> y el dato <i>\$action</i> con el que se identifica</p>

	<p>desde donde proviene la petición de comprobación.</p> <p>Primero se recogen todos los usuarios registrados en el curso en el <i>array</i> 'lista_usuarios'. Posteriormente se introducen en el <i>array</i> 'lista_idnumbers' los identificadores de los usuarios. Este <i>array</i> se usará para comprobar la existencia del usuario introducido en el sistema:</p> <ul style="list-style-type: none"> - Si el usuario introducido existe se introducirá en el vector 'miembros_grupos', comprobación que será distinta según el valor de <i>\$action</i>. - Si el usuario no existe no se introducirá en el vector <p>Una vez comprobado si los usuarios existen, se comprueba si hay algún identificador repetido. Para cada miembro del vector 'miembros_grupos' se introduce en el <i>array</i> 'contador' el número de veces que aparece.</p> <p>Ver <i>blended_clean_idnumber()</i></p>	
Argumentos	<i>\$members_group</i>	Array con los miembros introducidos en los grupos
	<i>\$action</i>	Constante que indica de donde proviene la petición ('saveteams.php' o 'teamsmanagement.php')
Return	Devuelve un contador con el número de repeticiones de cada miembro en los grupos	

Tabla 27: Función *blended_num_members*

<p>function blended_insert_teams(\$teams=null,\$grades=null,\$members,\$course,\$blended=null,\$leader=null,\$id_assignment=null,\$id_assignment_0=null,\$id=null,\$currentuserid=null,\$groupingid)</p>	
Descripción	<p>Inserta nuevos equipos en las tablas asociadas a los agrupamientos. Cada equipo consta de un objeto 'team', un objeto 'member' y un objeto 'grade'. Recibe el <i>array</i> unidimensional 'teams' con todos los objetos 'team' y otro bidimensional 'members' indexado por un índice entero secuencial que hace referencia a cada objeto 'team' y por los identificadores de los miembros asociados a cada 'team' y/u otro <i>array</i> unidimensional 'grades' con todos los objetos 'grade':</p> <ul style="list-style-type: none"> - Cuando recibe los <i>arrays</i> <i>\$teams</i> y <i>\$members</i>, inserta uno a uno cada objeto 'team' junto con el <i>array</i> de los identificadores de todos sus miembros asociados. Guarda en el <i>array</i> 'insertedteams' cada objeto <i>team</i> insertado, 'insertedteam' (y después recuperado para saber cuál es la clave primaria (id) asignada por Moodle y poder completar así los objetos 'member' y 'grade' antes de insertarlos). - Solo insertará aquellos equipos que posean algún miembro que exista en la base de datos puesto que los miembros que no existen no serán guardados. <p>Si hay error al insertar alguno de los elementos del <i>array</i> <i>\$members</i>:</p>

	<ul style="list-style-type: none"> - Se guarda en el <i>array</i> <i>\$array_return</i> indexado por el nombre del equipo, el código de error que contiene el tipo de error junto con el identificador de cada uno de los miembros que ha dado error. - Se borra el objeto <i>'team'</i> recién insertado y todos los elementos de <i>\$members</i> asociados al mismo que hayan sido insertados. - Cuando sólo recibe el <i>array</i> <i>\$grades</i> inserta uno a uno cada objeto <i>'grade'</i> en la tabla correspondiente. - Cuando recibe todos los <i>arrays</i>, primero inserta <i>\$teams</i> y <i>\$members</i> y genera el <i>array</i> <i>'insertedteams'</i> que lo utiliza posteriormente para completar cada objeto <i>'grade'</i> del <i>array</i> <i>\$grades</i> antes de insertarlos todos de uno en uno. <p>Ver:</p> <ul style="list-style-type: none"> - <i>blended_clean_idnumber()</i> - <i>blended_insert_team()</i> - <i>blended_add_new_group()</i> 	
Argumentos	<i>\$teams</i>	Array con los equipos
	<i>\$grades</i>	Array con las calificaciones
	<i>\$members</i>	Array con los miembros de cada equipo
	<i>\$course</i>	Objeto de la instancia del curso
	<i>\$blended</i>	Objeto de la instancia del módulo <i>Blended</i>
	<i>\$leader</i>	Valor que permite saber si un usuario es líder de su equipo
	<i>\$id_assignment</i>	Identificador de la actividad actual
	<i>\$id_assignment_0</i>	Identificador de la actividad de la cual se extrajeron los equipos
	<i>\$id</i>	Identificador del módulo del curso
	<i>\$currentuserid</i>	Objeto de la tabla "user" perteneciente al usuario que está usando el módulo
<i>\$groupingid</i>	Identificador del agrupamiento	
Return	Devuelve un array con los códigos de error o <i>null</i> si todo ha sido correcto	

Tabla 28: Función *blended_insert_teams*

function blended_insert_team (\$team=null,\$grade=null,\$insertedteam=null)	
Descripción	<p>Inserta un objeto <i>\$team</i> o un objeto <i>\$grade</i> en las tablas "blended_team" o "blended_grade" respectivamente:</p> <ul style="list-style-type: none"> - Cuando recibe un objeto <i>\$team</i> junto con el tercer parámetro <i>\$insertedteam</i> pasado por referencia, primero lo inserta en su tabla y después lo recupera de la misma para saber cuál es la clave primaria (<i>id</i>) que le ha asignado automáticamente <i>Moodle</i> y este objeto <i>\$team</i> se guarda en el objeto

	<p><i>\$insertedteam</i>.</p> <p>- Cuando recibe un objeto <i>\$grade</i> lo inserta en su tabla (si no existe ya en dicha tabla para el mismo equipo y la misma tarea, si existe se actualiza). Si ha recibido el tercer parámetro <i>\$insertedteam</i>, antes de ser insertado en la tabla se rellena su campo <i>\$id_team</i> con el <i>id</i> del objeto <i>\$insertedteam</i> recibido.</p> <p>Si se recibe una calificación de deberá comprobar que esa tarea y ese grupo existen y entonces modificar la tabla “blended_grade” borrando la calificación que pudiera haber e introduciendo una nueva.</p>	
Argumentos	<i>\$team</i>	Objeto ‘ <i>team</i> ’
	<i>\$grade</i>	Objeto ‘ <i>grade</i> ’
	<i>\$insertedteam</i>	Objeto ‘ <i>team</i> ’ recién insertado para obtener de él la clave primaria asignada por <i>Moodle</i>
Return	Devuelve <i>null</i> si todo es correcto o un mensaje de error si se ha producido algún problema	

Tabla 29: Función *blended_insert_team*

function blended_delete_sign_member_or_group (\$idteam,\$member, \$option)		
Descripción	<p>Borrado de un grupo completo o de un miembro de un grupo específico. Mediante el parámetro <i>\$option</i> se decide:</p> <ul style="list-style-type: none"> - Si <i>\$option</i>=‘<i>member</i>’ se borra el miembro dado por <i>\$idmember</i> del equipo con identificador <i>\$idteam</i>. - Si <i>\$option</i>=‘<i>group</i>’ se borra el equipo con identificador <i>\$idteam</i>. <p>Ver:</p> <ul style="list-style-type: none"> - <i>groups_remove_member()</i> - <i>groups_delete_group()</i> - <i>groups_unassign_grouping()</i> - <i>groups_delete_grouping()</i> 	
Argumentos	<i>\$idteam</i>	Identificador del equipo
	<i>\$member</i>	Objeto de la instancia de un miembro
	<i>\$option</i>	<i>String</i> que elige cual es la acción a realizar

Tabla 30: Función *blended_delete_sign_member_or_group*

function blended_comprobar_calificacion (\$id,\$instance,\$itemmodule, \$option=null)		
Descripción	<p>Obtención de las calificaciones de las tablas “blended_grade” y “grade_grades”.</p> <p>Se obtienen de ambas tablas aquellos alumnos que poseen calificaciones y se introducen en un vector sus <i>ids</i> y sus calificaciones (<i>arrays</i> distintos para cada tabla). Según el valor de <i>\$option</i> se devolverán :</p> <ul style="list-style-type: none"> - Si <i>\$option='blended'</i> el <i>array</i> de calificaciones de la tabla "blended_grade" - Si <i>\$option='grade'</i> el <i>array</i> de calificaciones de la tabla "grade_grades" <p>Mediante esta función se comprueba si las calificaciones de ambas tablas coinciden.</p>	
Argumentos	<i>\$id</i>	Identificador de la actividad
	<i>\$instance</i>	Instancia del objeto
	<i>\$itemmodule</i>	Módulo del objeto
	<i>\$option</i>	<i>String</i> que elige cual es la tabla a devolver
Return	Devuelve el array con las calificaciones de una de las tablas según el valor de <i>\$option</i> o <i>null</i> en el caso de que no existan calificaciones a devolver	

Tabla 31: Función *blended_comprobar_calificacion*

function blended_is_not_active_student (\$userid,\$course)		
Descripción	<p>Comprueba si un estudiante está inactivo en el curso, es decir si no ha entrado todavía. Se obtiene en el array <i>\$alluserids</i> todos los estudiantes matriculados mediante <i>blended_get_course_students_ids()</i> y el parámetro <i>\$selectactivestudents</i> igual a <i>false</i>. Se obtiene en el array <i>\$activeuserids</i> todos los estudiantes activos mediante <i>blended_get_course_students_ids()</i> y el parámetro <i>\$selectactivestudents</i> igual a <i>true</i>.</p> <p>Con la función <i>array_diff</i> se guarda en el array <i>\$noactivemembers</i> todos los valores de <i>\$alluserids</i> que no aparezcan <i>\$activeuserids</i>, es decir, los estudiantes inactivos. Por último se comprueba si el parámetro <i>\$userid</i> está en el array <i>\$noactivemembers</i>.</p> <p>Ver: <i>blended_get_course_students_ids()</i></p>	
Argumentos	<i>\$userid</i>	Identificador del objeto ‘ <i>user</i> ’ del miembro del equipo
	<i>\$course</i>	Objeto de la instancia curso
Return	Devuelve un booleano a <i>true</i> si el estudiante está inactivo o <i>null</i> si está activo	

Tabla 32: Función *blended_is_not_active_student*

function blended_get_teams_members (\$id_team=null, \$id_assignment=null, \$currentuser, \$blended, \$members, \$membersignedup='false', \$onlyidmember=false)															
Descripción	<p>Obtiene todos los miembros de un equipo o bien todos los miembros de todos los equipos que se están utilizando en la tarea actual. Cada miembro es representado por un objeto que contiene campos de su objeto <i>'user'</i> y todos los campos de su objeto <i>'member'</i>.</p> <p>Tres modos de funcionamiento:</p> <p>1: Si recibe <i>\$id_team</i> devuelve un <i>array</i> unidimensional con todos los miembros de un equipo. <i>Array</i> indexado por la clave primaria de la tabla <i>member</i>.</p> <p>2: Si recibe <i>\$id_assignment</i> devuelve un <i>array</i> bidimensional con todos los equipos con todos sus miembros para una asignatura determinada. <i>Array</i> indexado por la clave primaria de la tabla "groups" y por la clave primaria de la tabla "blended_assign_grouping".</p> <p>3: Si recibe <i>\$onlyidmember</i> igual a <i>true</i> devuelve un <i>array</i> unidimensional con todos los identificadores (<i>'idmember'</i>) de los miembros del equipo <i>\$id_team</i>.</p> <p>También recibe un <i>string</i> (<i>\$membersignedup</i>) pasado por referencia con el valor por defecto <i>'false'</i> y el objeto <i>'user'</i> del usuario que actualmente está utilizando el modulo (<i>\$currentuser</i>) para saber si el usuario actual está inscrito en alguno de los equipos, en cuyo caso el <i>string</i> se pone al valor <i>'true'</i>.</p> <p>Ver: <i>blended_gen_cleanidvalue()</i></p>														
Argumentos	<table border="1"> <tbody> <tr> <td><i>\$id_team</i></td> <td>identificador del equipo</td> </tr> <tr> <td><i>\$id_assignment</i></td> <td>identificador de la tarea</td> </tr> <tr> <td><i>\$currentuser</i></td> <td>objeto <i>'user'</i> del usuario actual del módulo</td> </tr> <tr> <td><i>\$blended</i></td> <td>objeto del módulo blended</td> </tr> <tr> <td><i>\$members</i></td> <td><i>array</i> donde se guardarán los objetos <i>user</i> de los miembros del equipo. Unidimensional o bidimensional dependiendo del modo de funcionamiento</td> </tr> <tr> <td><i>\$membersignedup</i></td> <td><i>flag</i> para saber si el usuario actual del módulo está inscrito en algún equipo para la tarea</td> </tr> <tr> <td><i>\$onlyidmember</i></td> <td><i>true</i> si sólo se quiere recuperar un <i>array</i> con los identificadores (en vez del objeto <i>'user'</i>) de los miembros del/los equipo/s</td> </tr> </tbody> </table>	<i>\$id_team</i>	identificador del equipo	<i>\$id_assignment</i>	identificador de la tarea	<i>\$currentuser</i>	objeto <i>'user'</i> del usuario actual del módulo	<i>\$blended</i>	objeto del módulo blended	<i>\$members</i>	<i>array</i> donde se guardarán los objetos <i>user</i> de los miembros del equipo. Unidimensional o bidimensional dependiendo del modo de funcionamiento	<i>\$membersignedup</i>	<i>flag</i> para saber si el usuario actual del módulo está inscrito en algún equipo para la tarea	<i>\$onlyidmember</i>	<i>true</i> si sólo se quiere recuperar un <i>array</i> con los identificadores (en vez del objeto <i>'user'</i>) de los miembros del/los equipo/s
<i>\$id_team</i>	identificador del equipo														
<i>\$id_assignment</i>	identificador de la tarea														
<i>\$currentuser</i>	objeto <i>'user'</i> del usuario actual del módulo														
<i>\$blended</i>	objeto del módulo blended														
<i>\$members</i>	<i>array</i> donde se guardarán los objetos <i>user</i> de los miembros del equipo. Unidimensional o bidimensional dependiendo del modo de funcionamiento														
<i>\$membersignedup</i>	<i>flag</i> para saber si el usuario actual del módulo está inscrito en algún equipo para la tarea														
<i>\$onlyidmember</i>	<i>true</i> si sólo se quiere recuperar un <i>array</i> con los identificadores (en vez del objeto <i>'user'</i>) de los miembros del/los equipo/s														
Return	Devuelve <i>true</i> si el <i>array</i> se ha obtenido correctamente y <i>false</i> si error														

Tabla 33: Función *blended_get_teams_members*

function blended_get_teams_members_from_grades (\$id_assignment, \$currentuser, \$members, \$membersignedup='false')									
Descripción	<p>Obtiene todos los miembros de los equipos que se están utilizando en la tarea actual pero que fueron definidos en otra tarea. Cada miembro es representado por un objeto que contiene campos de su objeto <i>'user'</i> y campos de otras tablas.</p> <p>Recibe un array bidimensional vacío pasado por referencia (equipos para la tarea <i>\$id_assignment</i> x miembros de cada equipo) en donde se guardan indexados por el identificador de cada equipo todos los objetos <i>'user'</i> pertenecientes a los miembros de cada equipo.</p> <p>La diferencia con la función anterior es que los equipos definidos para la tarea <i>\$id_assignment</i> son los mismos que los que se crearon para otra tarea anterior (<i>\$id_assignment_0</i>), por eso el identificador de equipo <i>id_team</i> lo obtenemos de la tabla "blended_grade" y no de la tabla "blended_team". También se accede a la tabla "assignment" para obtener el nombre de la tarea (<i>name_assignment_0</i>) para la cual fueron creados originalmente los equipos que ahora se utilizarán para la tarea <i>\$id_assignment</i>.</p> <p>Si se recibe el objeto <i>\$currentuser</i> se comprueba si el estudiante que llama a esta función ya está inscrito en un equipo para esa tarea, en tal caso se activa el <i>flag \$membersignedup</i></p>								
Argumentos	<table border="1"> <tr> <td><i>\$id_assignment</i></td> <td>identificador de la tarea</td> </tr> <tr> <td><i>\$currentuser</i></td> <td>objeto <i>'user'</i> del usuario actual del módulo</td> </tr> <tr> <td><i>\$members</i></td> <td>array donde se guardarán los objetos <i>user</i> de los miembros del equipo</td> </tr> <tr> <td><i>\$membersignedup</i></td> <td>flag para saber si el usuario actual del módulo está inscrito en algún equipo para la tarea</td> </tr> </table>	<i>\$id_assignment</i>	identificador de la tarea	<i>\$currentuser</i>	objeto <i>'user'</i> del usuario actual del módulo	<i>\$members</i>	array donde se guardarán los objetos <i>user</i> de los miembros del equipo	<i>\$membersignedup</i>	flag para saber si el usuario actual del módulo está inscrito en algún equipo para la tarea
<i>\$id_assignment</i>	identificador de la tarea								
<i>\$currentuser</i>	objeto <i>'user'</i> del usuario actual del módulo								
<i>\$members</i>	array donde se guardarán los objetos <i>user</i> de los miembros del equipo								
<i>\$membersignedup</i>	flag para saber si el usuario actual del módulo está inscrito en algún equipo para la tarea								
Return	Devuelve <i>true</i> si el array se ha obtenido correctamente y <i>false</i> si error								

Tabla 34: Función *blended_get_teams_members_from_grades*

Function blended_get_name_team (\$id_assignment, \$r)					
Descripción	Obtiene el nombre de equipo disponible para la tarea actual. Por defecto a los equipos se les asigna un nombre que consiste en un entero que se incrementa secuencialmente. Busca cual es el primer entero libre en la secuencia (que comienza desde 1) en el campo <i>`name_team`</i> de la tabla "blended_team" para la tarea dada para asignarlo como nombre de equipo.				
Argumentos	<table border="1"> <tr> <td><i>\$id_assignment</i></td> <td>Identificador de la actividad</td> </tr> <tr> <td><i>\$r</i></td> <td>Guardaré el primer entero libre en la secuencia.</td> </tr> </table>	<i>\$id_assignment</i>	Identificador de la actividad	<i>\$r</i>	Guardaré el primer entero libre en la secuencia.
<i>\$id_assignment</i>	Identificador de la actividad				
<i>\$r</i>	Guardaré el primer entero libre en la secuencia.				
Return	Devuelve el nombre del equipo				

Tabla 35: Función *blended_get_name_team*

function blended_get_assignment_name (\$id_assign)	
Descripción	Obtiene el nombre de la actividad a partir de su identificador. Ver <i>get_coursemodule_from_id()</i>
Argumentos	<i>\$id_assign</i> identificador de la actividad
Return	Devuelve el nombre de la actividad

Tabla 36: Función *blended_get_assignment_name*

function blended_get_user (\$id_member=null, \$blended=null, \$userid=null)	
Descripción	Obtiene el objeto ' <i>user</i> ' asociado a un miembro de un equipo a partir de: - su identificador <i>EAN-13</i> (<i>id</i> o <i>idnumber</i> de la tabla " <i>user</i> " o bien el campo personalizado de la tabla " <i>user_info_data</i> "). - a partir de su <i>id</i> de la tabla " <i>user</i> " directamente.
Argumentos	<i>\$id_member</i> identificador del miembro del equipo
	<i>\$blended</i> objeto del módulo <i>Blended</i>
	<i>\$userid</i> identificador del objeto ' <i>user</i> ' del miembro del equipo
Return	Devuelve objeto ' <i>user</i> ' o <i>null</i> si error

Tabla 37: Función *blended_get_user*

function blended_get_grades (\$id_team, \$id_assignment, \$alert)	
Descripción	Obtiene la calificación de un equipo para una determinada tarea. Primero se obtiene la nota del grupo de la tabla " <i>blende_grade</i> ". Después se compara con cada una de las notas individuales de los miembros del equipo obtenidas del libro de calificaciones. Si a algún miembro del equipo se le ha modificado su nota individual (en el libro de calificaciones), resultando distinta a la de grupo (guardada en " <i>blended_grade</i> ") se activa el flag <i>\$alert</i> pasado por referencia. Si se cambian manualmente las notas individuales de varios miembros del equipo en el libro de calificaciones a un mismo valor, y el número de estos miembros con su nota modificada es mayor que la mitad de los miembros del equipo esta nueva nota pasa a ser la nueva nota global del equipo en la tabla " <i>blended_grade</i> ". Ver: <i>blended_count_repeat_values()</i>

Argumentos	<i>\$id_team</i>	identificador de objeto 'team'
	<i>\$id_assignment</i>	identificador de la actividad
	<i>\$alert</i>	Posee el valor <i>true</i> si alguna nota individual de los miembros del equipo es distinta a la de grupo
Return	Devuelve la nota de grupo	

Tabla 38: Función *blended_get_grades*

function blended_count_repeat_values (\$grade, \$grades_array, \$steambookgrade)		
Descripción	Comprueba si ha habido alguna modificación de la calificación individual de un miembro de un equipo en el <i>gradebook</i> .	
	Recibe un <i>array</i> con todas las calificaciones individuales (obtenidas del libro de calificaciones de <i>Moodle</i>) de todos los miembros del equipo y comprueba si todas ellas son iguales a la del equipo (obtenido de la tabla "blended_grade").	
	Si se cambian manualmente las notas individuales de varios miembros del equipo en el libro de calificaciones a un mismo valor, y el número de estos miembros con su nota modificada es mayor que la mitad de los miembros del equipo esta nueva nota pasa a ser la nueva nota global del equipo en la tabla " <i>blended_grade</i> ".	
Argumentos	<i>\$grade</i>	nota del equipo obtenida de la tabla " <i>blended_grade</i> "
	<i>\$grades_array</i>	<i>array</i> con todas las notas individuales de los miembros del equipo obtenidas del libro de calificaciones
	<i>\$steambookgrade</i>	guarda la posible nueva nota del grupo
Return	Devuelve si la nota individual ha sido modificada y <i>false</i> si todas las notas de los miembros son iguales a la de equipo	

Tabla 39: Función *blended_count_repeat_values*

function blended_get_grademax (\$id_assignment)		
Descripción	Obtiene la calificación máxima que se puede dar a una determinada tarea. Por defecto la máxima calificación es 10.	
	Ver <i>grade_get_grade_items_for_activity()</i>	
Argumentos	<i>\$id_assignment</i>	identificador de objeto 'assignment'
Return	Devuelve la nota máxima	

Tabla 40: Función *blended_get_grademax*

function blended_get_error_alert (\$array_return)			
Descripción	<p>Muestra por pantalla los mensajes de error cuando ha ocurrido algún error al insertar nuevos miembros de equipo en las tablas. Hay tres tipos de errores:</p> <ul style="list-style-type: none"> - Estudiante con identificador no válido. - Estudiante no matriculado en el curso. - Estudiante ya inscrito en otro equipo para la misma tarea. <p>La función genera un <i>string</i> donde se listan clasificados por cada uno de los tres posibles errores los identificadores de aquellos miembros de equipo y a que equipo pertenecen que hayan ocasionado el error.</p> <p>Se hace control del número de equipos y el número de miembros por equipo que han ocasionado error para generar las distintas frases con la correcta concordancia de singular/plural (entre 'sujetos' y 'verbos').</p>		
Argumentos	<table border="1"> <tr> <td><i>\$array_return</i></td> <td>array bidimensional indexado por los identificadores de los equipos que han dado error al ser insertados junto con los código de error de cada miembro del equipo que ha ocasionado algún tipo de error</td> </tr> </table>	<i>\$array_return</i>	array bidimensional indexado por los identificadores de los equipos que han dado error al ser insertados junto con los código de error de cada miembro del equipo que ha ocasionado algún tipo de error
<i>\$array_return</i>	array bidimensional indexado por los identificadores de los equipos que han dado error al ser insertados junto con los código de error de cada miembro del equipo que ha ocasionado algún tipo de error		
Return	Devuelve mensaje que se mostrará por pantalla con la lista de todos los errores que han ocurrido al insertar los miembros de equipo		

Tabla 41: Función *blended_get_error_alert*

function pdfAssignmentPage (\$code, \$margins, \$fullname \$assignment,\$course,\$blended,\$gradingmarks=false)	
Descripción	Función encargada de devolver la página de tarea de usuario.
Return	Devuelve por pantalla el PDF generado.

Tabla 42: Función *pdfAssignmentPage*

function blended_crear_equipo_por_alumno (\$cm,\$blended,\$id_assignment, \$name_team,\$course,\$idvalue,\$id_currentuser)	
Descripción	Función que permite a un alumno crear un nuevo equipo dentro de un agrupamiento ya existente. Se realiza la comprobación de que no se encuentra inscrito en otro equipo para la misma tarea ya que de ser así no podría crearlo. También se comprueba si el alumno tiene la posibilidad de crearlo,

	cuya posibilidad se la concede el profesor.	
	Ver:	
	<ul style="list-style-type: none"> - <i>blended_get_name_team()</i> - <i>groups_get_all_groups()</i> - <i>blended_update_grouping()</i> 	
Argumentos	<i>\$cm</i>	Objeto instancia de la actividad
	<i>\$blended</i>	Objeto instancia del módulo <i>Blended</i>
	<i>\$id_assignment</i>	Identificador de la actividad
	<i>\$name_team</i>	Nombre del equipo
	<i>\$course</i>	Objeto de la instancia del curso
	<i>\$idvalue</i>	Código de identificación del alumno
	<i>\$id_currentuser</i>	Identificador del usuario actual

Tabla 43: Función *blended_crear_equipo_por_alumno*

function blended_crear_nuevo_agrupamiento_por_alumno (\$blended, \$id_assignment, \$name_team,\$course,\$idvalue,\$id_currentuser)		
Descripción	<p>Función que permite a un alumno crear un nuevo equipo en un nuevo agrupamiento que también crea. Se realiza la comprobación de que no se encuentra inscrito en otro equipo para la misma tarea ya que de ser así no podría crearlo. También se comprueba si el alumno tiene la posibilidad de crearlo, cuya posibilidad se la concede el profesor.</p> <p>Ver:</p> <ul style="list-style-type: none"> - <i>get_coursemodule_from_instance()</i> - <i>groups_create_grouping()</i> - <i>blended_add_new_group()</i> - <i>blended_update_teams_by_groups</i> 	
Argumentos	<i>\$blended</i>	Objeto instancia del módulo <i>Blended</i>
	<i>\$id_assignment</i>	Identificador de la actividad
	<i>\$name_team</i>	Nombre del equipo
	<i>\$course</i>	Objeto de la instancia del curso
	<i>\$idvalue</i>	Código de identificación del alumno
	<i>\$id_currentuser</i>	Identificador del usuario actual

Tabla 44: Función *blended_crear_nuevo_agrupamiento_por_alumno*

function blended_inscribir_equipo_por_alumno (\$cm,\$blended,\$id_assignment, \$id_team,\$course,\$idvalue,\$currentuser)	
Descripción	Función que permite a un alumno inscribirse en un equipo ya existente. Se

	realiza la comprobación de que no se encuentra inscrito en otro equipo para la misma tarea ya que de ser así no podría inscribirse.	
	Ver:	
	<ul style="list-style-type: none"> - <code>groups_get_all_groups()</code> - <code>blended_update_grouping()</code> 	
Argumentos	<code>\$cm</code>	Objeto instancia de la actividad
	<code>\$blended</code>	Objeto instancia del módulo <i>Blended</i>
	<code>\$id_assignment</code>	Identificador de la actividad
	<code>\$id_team</code>	Identificador del equipo
	<code>\$course</code>	Objeto de la instancia del curso
	<code>\$idvalue</code>	Código de identificación del alumno
	<code>\$currentuser</code>	Identificador del usuario actual

Tabla 45: Función *blended_inscribir_equipo_por_alumno*

4.4.2. *Ficheros de Blended*

Fichero labels.php

A este fichero se accede desde *view.php* y está encargado de mostrar al usuario ya sea profesor o alumno las opciones a escoger para poder obtener las etiquetas de un alumno por pantalla.

El permiso de ejecución del fichero viene dado en el fichero *access.php* como:

```
'mod/blended:printlabels' => array (
    'captive' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_ALLOW,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

Para todos los usuarios con permisos en este fichero se le mostrarán una serie de opciones para poder elegir el formato de la página de etiquetas que desea obtener. Entre las opciones se encuentran el tipo de hoja a imprimir, los márgenes de la misma y el número de filas y de columnas de etiquetas que se desean imprimir. Todas estas opciones se encuentran dentro del formulario *rowscolumnsform* y poseen una serie de valores por defecto. El número de filas y de columnas se muestran como un desplegable cuyos valores máximos vienen dados por *ROWS_MAX_ENTRIES* y *COLUMNS_MAX_ENTRIES*.

El resto del fichero posee distintas opciones según el tipo de usuario. Para poder diferenciar esto, se utilizan los permisos de *selectamongallstudents* definido en *access.php*:

```
'mod/blended:selectoneamongallstudents' => array (
    'capytype' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_PREVENT,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

Por tanto habrá distintos vínculos si el usuario es un profesor o un alumno ya que si es profesor, éste tendrá la capacidad de elegir a los alumnos de los que desea imprimir sus páginas de etiquetas. Para poder realizar esto se obtienen los identificadores de los alumnos inscritos en el curso con la función *blended_get_course_student_ids()*. Para cada usuario se comprobará si está activo en el curso mediante *blended_is_not_active_student()* y si posee identificador de usuario con *blended_gen_idvalue()*. Si alguna de estas condiciones no se cumple se mostrará por pantalla un aviso al profesor. Tras realizar estas comprobaciones se muestra al profesor una tabla con los nombres de los alumnos.

El profesor tiene la capacidad de elegir que etiquetas desea obtener, si una por cada usuario activo, páginas completas para todos los usuarios elegidos o una por cada estudiante matriculado en el curso.

Por último a todos los usuarios se les dan las opciones de elegir el contenido de la etiqueta: sin identificador, con identificador de usuario o con el nombre completo del alumno.

Al final de la página se muestra el botón *imprimir etiquetas* accediendo a la siguiente página: *printCourseLabels.php*

Fichero printCourseLabels.php

A este fichero se accede desde *labels.php* y posee los mismos permisos que éste.

Mediante la función *blended_gen_idvalue()* se comprueba si el usuario posee código de identificación ya que si no es así se mostrará un mensaje de que no posee identificador y por tanto no se podrá imprimir su página de etiquetas.

Tras realizar esta comprobación se recoge el valor del número de filas *numrows* y de columnas *numcolumns*. Se comprueba si el valor recibido se encuentra entre los valores posibles ya que si no es así se mostrará un mensaje de aviso y se volverá a *labels.php*.

Si el valor de *numrows* y *numcolumns* es correcto se recoge que tipo de estudiantes ha elegido el profesor para imprimir sus etiquetas: activos, todos o uno en concreto.

Para poder imprimir las etiquetas se llama a la función *pdfLabelsPage()*.

Fichero assignmentpage.php

A este fichero se accede desde *view.php* para aquellos usuarios que deseen imprimir una página de una de las tareas creadas en el curso. Los permisos para poder acceder a esta página vienen dados en *access.php* por

```
'mod/blended:printassignmentpage' => array (
    'captive' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_ALLOW,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

Por tanto, a este fichero pueden acceder tanto profesores como alumnos matriculados.

Este fichero está formado por el formulario *assignmentform* cuyo contenido será distinto dependiendo del usuario.

Para cualquier usuario se mostrará una lista desplegable con las tareas creadas en el curso que sean de los módulos *assign* o *assignment*. Mediante este desplegable se podrá elegir la tarea de la que se desea imprimir su página.

Si el usuario es un profesor, éste tendrá la capacidad de elegir al estudiante del que desea imprimir la página de tarea. Para ello se hará uso de los permisos dados por *selectamongallstudents*. De la misma manera que en el fichero *labels.php* se recogerán los usuarios y se mostrarán por pantalla los usuarios del curso avisando de si alguno de ellos no está activo o no posee identificador de usuario.

Por último se muestra el botón *Imprimir página de tarea* que permite entrar en el fichero *printassignmentpage.php*.

Fichero printassignmentpage.php

A este *script* se accede desde *assignmentpage.php* y posee los mismos permisos que éste.

Para poder obtener por pantalla la página de tarea, primero se comprueba si la tarea existe en la base de datos y si no es así entonces se mostrará un mensaje de error mediante *print_error()*.

Del fichero *assignmentpage.php* se recoge la variable *\$code* que contiene un valor que indica si el usuario posee o no código de identificación en el sistema, ya que si no lo posee no se podrá imprimir la página de tarea avisando al usuario mediante la función *notice()* de la situación y volviendo a *assignmentpage.php*.

Si todo es correcto se llamará a la función *pdfassignmentpage()* encargada de mostrar por pantalla la página de tarea en la que se mostrarán no solo la etiqueta del usuario sino también las características de la tarea como la fecha de entrega, el nombre y la descripción.

Fichero teamsmanagement.php

A este *script* se accede desde *view.php* y está encargado de mostrar las actividades creadas en el curso que permiten realizar grupos de trabajo. Posee los permisos dados por *access.php* como

```
'mod/blended:introteams' => array (
    'capytype' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_PREVENT,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

Con estos permisos ni los invitados ni los alumnos pueden entrar, sólo es accesible a administradores y profesores.

En esta página se muestra en una tabla mediante `html_writer::table($table)` el objeto `$table` con todas las actividades del curso que son calificables y que puedan poseer grupos de trabajo. Por tanto, primero se recoge la lista de estas actividades mediante `new grade_tree()`.

Mediante `get_coursemodules_in_course()` se obtiene la información de cada una de las actividades listadas como por ejemplo la fecha de entrega de la actividad.

En la tabla mostrada en la página aparecerá:

- La fecha de entrega de la actividad: el formato viene dado por `userdate()`. Si la actividad no posee esta característica el cuadro aparecerá vacío.
- Número de equipos definidos:
 - Si hay equipos definidos, se mostrará:
 - Número de equipos: obtenido de las tablas de los agrupamientos y guardado en la variable `$numteams`.
 - Enlace a 'modificar equipos': este enlace dado por `$teamlink1` permite ir al `script updateTeams.php` donde se pueden modificar los equipos generados.
 - Enlace a 'inicializar equipos': este enlace dado por `$teamlink` permite inicializar los equipos generados asignando a la actividad un nuevo agrupamiento distinto al que poseía y por tanto con grupos diferentes dado por el `script introteams.php`.
 - Si no hay equipos definidos
 - El cuadro de número de equipos se mostrará vacío.
 - Enlace a 'inicializar equipos' dado por `$teamlink`.
- Cuadro donde se muestra si la tarea está calificada. La comprobación de esta característica se guarda en el objeto `$grades`. Si no hay equipos definidos la variable `$grade` poseerá valor nulo y se mostrará el cuadro vacío. Si hay equipos, se mostrará si está calificada (Si) cuando el número de equipos calificados dado por `count($grades)` sea igual al número de equipos definidos `$numteams`. Si son distintos mostrará parcialmente (parcialmente) y si `$grades` no posee valor ya que no hay ninguno que esté calificado mostrará que no lo está (No). Todos ellos son un enlace, cuyo valor se guarda en la variable `$gradelink`, al `script` que permite calificarla `introgrades.php`.

Fichero introteams.php

A este *script* se accede desde *teamsmanagement.php* o desde *grades.php* y posee los mismos permisos de acceso que éstos.

Cuando se accede a este enlace, se pretende inicializar los grupos de una actividad. Toda actividad que se desee que posea grupos debe poseer antes un agrupamiento ya que es éste el que posee los grupos. A este enlace se puede acceder cuando aún no hay equipos o cuando los hay pero se desea reinicializar la actividad, es por ello por lo que si la actividad poseía grupos calificados, estas calificaciones deben ser borradas tanto de la tabla *'blended_grade'* como de *'grade_grades'* borrándose así tanto de las tablas de *blended* como del libro de calificaciones.

En este *script* se da al usuario dos posibilidades de inicializar los grupos:

1. Elegir un agrupamiento existente: En el curso puede haber agrupamientos existentes que posean grupos y por tanto se da al usuario la posibilidad de elegir un agrupamiento asignando esos grupos ya creados a la actividad. Para ello, primero se comprueba si existe algún agrupamiento en el curso mediante *groups_get_all_groupings()*. Los agrupamientos obtenidos se recogen en *\$groupings*. Se recorre mediante un bucle *foreach()* estos agrupamientos obteniendo el nombre que poseen para poder mostrarlo en *\$nameagr*. Mediante un desplegable dentro del formulario *fl* se mostrarán los nombres de los agrupamientos y el botón *continuar* que permite que se realice la operación dirigiéndose a *updateTeams.php*.
2. Crear un nuevo agrupamiento: Se le muestra al usuario la posibilidad de crear un nuevo agrupamiento con nuevos grupos de manera manual. Mediante el formulario *signupform* se muestra una tabla donde se le da al usuario distintas opciones:
 - Método de creación: selección de tipo de método de creación de equipos que desea realizar: manual o aleatoria guardado en *\$creationmethod*.
 - Si el usuario escoge manual tendrá que introducir a mano uno a uno los alumnos que desee.
 - Si por el contrario elige aleatorio se activará un desplegable que permita elegir que alumnos desea introducir: los activos o todos. Se asignarán de manera aleatoria los alumnos elegidos en los distintos grupos. Si la elección se escoge manual el desplegable permanecerá inactivo. Esto se da mediante la función *enableSelectStudents()*.
 - Número de equipos que desea crear : *'numteams'*
 - Número de miembros por equipo: *'nummembers'*.
 - Nombre del agrupamiento que se va a crear: *'name_team'*

Por último, se muestra el botón 'Nuevo agrupamiento' que dirige la creación de los equipos con los parámetros definidos al *script createTeams.php*.

Fichero updateTeams.php

A este *script* se accede desde *introTeams.php*, desde *teamsmanagement.php*, desde *saveteams.php* o desde *update_groupings.php* y posee los mismos permisos que el *script teamsmanagement.php*.

A este *script* se accede para poder modificar los grupos del agrupamiento asignado a una actividad y por tanto lo primero que se realiza es obtener el nombre de la actividad mediante la función *blended_get_assignmnet_name()*.

Si la acción proviene de *introTeams.php* significa que se ha elegido asignar un agrupamiento existente a la actividad y por tanto se van a obtener los grupos de ese agrupamiento mediante *groups_get_all_groups()* guardándolos en *\$groups_groupings* y sus miembros mediante *groups_get_members()* guardados en *\$groups_members*. Si no existen grupos se mostrará que el agrupamiento está vacío volviendo al *script teamsmanagement.php*.

Si la acción proviene de *update_groupings.php* implica que se han modificado miembros de los grupos y por tanto se debe realizar la comprobación de si existen miembros repetidos en los grupos. Para esto, se recogen en *\$repetir_miembro* el número de veces que aparece cada alumno en el agrupamiento.

Si la acción proviene de *teamsmanagement.php* se recoge en *\$groups_groupings_g* los grupos del agrupamiento que posee asignada la actividad mediante la función *groups_get_all_groups()* y sus miembros en *\$grouping_members* mediante la función *groups_get_members()*. Se recoge en *\$repetir_miembro()* cuantas veces aparece cada alumno mediante la función *blended_num_members()*.

Cada vez que se entra en este *script* se actualizan los grupos y miembros de *blended* con los datos de los agrupamientos para así poder tener las tablas sincronizadas. Esto es realizado mediante la función *blended_update_teams_by_groups()*.

Puede darse el caso de que haya equipos que estén calificados y es por ello por lo que sus calificaciones deben ser actualizadas ya que puede haber miembros que se hayan borrado, cambiados de grupo o introducidos. Para realizar esto se llaman a las funciones *grade_update()* y *update_final_grade()*.

Mediante este *script* se pueden actualizar los grupos del agrupamiento de dos maneras distintas:

1. Reinicializar los grupos y sus miembros: Si el profesor desea reinicializar todos los grupos, tiene la posibilidad de cambiar todos los grupos del agrupamiento. Mediante el formulario *signupform* se muestra una tabla con las siguientes opciones:
 - Método de creación: selección de tipo de método de creación de equipos que desea realizar: manual o aleatoria guardado en *\$creationmethod*.
 - Si el usuario escoge manual tendrá que introducir a mano uno a uno los alumnos que desee.
 - Si por el contrario elige aleatorio se activará un desplegable que permita elegir que alumnos desea introducir: los activos o todos. Se asignarán de manera aleatoria los alumnos elegidos en los distintos grupos. Si la elección se escoge manual el desplegable permanecerá inactivo. Esto se da mediante la función *enableSelectStudents()*.
 - Número de equipos que desea crear : *'numteams'*
 - Número de miembros por equipo: *'nummembers'*

Por último, se muestra el botón *'Reinicializar grupos'* que dirige la reinicialización de los equipos con los parámetros definidos al *script createTeams.php*.

2. Actualizar pequeños cambios en los grupos del agrupamiento: Mediante el formulario *teamsform* se muestran los grupos y los miembros de cada grupo dentro del agrupamiento que tiene asignado la actividad. Los equipos se recogen mediante un bucle *foreach* con la función *groups_get_group()* almacenándolos en la variable *\$g* , mostrando para cada grupo un cuadro con el nombre que posee, y los miembros de cada grupo se recogen con la función *groups_get_members()* almacenándolos en la variable *\$mem*. En cada grupo se recorren mediante un bucle *foreach* cada uno de los miembros que posee mostrándose cada uno en un cuadro con su código de identificación obtenido mediante la función *blended_gen_idvalue()*. Además de su código de identificación se muestra al lado de cada usuario su foto de perfil almacenada en *\$userpic*.

Puede darse el caso de que un profesor pueda introducir sin darse cuenta un mismo usuario en dos grupos distintos, es por ello por lo que para cada miembro se comprueba cuantas veces aparece de tal forma que si aparece repetido se recuadrará en rojo su código de identificación avisando al profesor de la situación.

Para un mejor manejo de los equipos se da al profesor la posibilidad de añadir un miembro más a cada equipo o un equipo más al agrupamiento.

Tras mostrar la tabla con los equipos, se muestra el botón '*Actualizar Agrupamiento*' que permite actualizar los cambios introducidos dirigiéndose al *script update_groupings.php*.

También se da la posibilidad al usuario de volver a la página *teamsmanagement.php* mediante el botón '*Ir a: Administración de Equipos*'.

Fichero update_groupings.php

A este *script* se accede desde *updateTeams.php* y posee los mismos permisos que éste. Mediante este *script* se pueden actualizar los cambios realizados en los grupos de un agrupamiento que tiene asociada una actividad.

Se recoge del formulario '*teamsform*' de *updateTeams.php* el número de equipos *numteams* y el número de miembros por equipos *nummembers*. Mediante un bucle *for* se recoge cada equipo mediante *\$_POST* obteniendo del formulario el nombre de cada equipo almacenándolo en *\$teamname*, el identificador de cada equipo en *\$id_team* y mediante otro bucle *for* se recorren los miembros obteniendo sus códigos de identificación y guardándolos en *\$memberid*.

Con los parámetros obtenidos se actualizarán los equipos del agrupamiento mediante la función *blended_actualizar_agrupamiento()* de donde se recoge el contador de las veces que aparece cada miembro almacenándolo en *\$contador*.

Tras realizar la actualización se redirige la página a *updateTeams.php* mediante el formulario *updategrouping*.

Fichero createTeams.php

A este *script* se accede desde *introTeams.php* o desde *updateTeams.php* y posee los mismos permisos que estos.

Este *script* permite la creación de nuevos equipos en un agrupamiento. La estructura de la página depende de los parámetros pasados: número de equipos, número de miembros por equipo y método de creación. Se mostrarán los equipos mediante cuadros en los que deben aparecer los códigos de identificación de los alumnos a la hora de guardar los equipos. A priori, los nombres de los equipos serán números empezando desde el 1 hasta *numteams*. Si el método de creación es aleatorio, mediante la función *blended_method_random()* se recogerán los miembros llamados según el valor de *studentsselection* : activos o todos, guardado en el array *\$freemembers*. Si no hay

ningún estudiante activo o matriculado en el curso se mostrará un mensaje por pantalla y se volverá a *teamsmanagement.php*.

Si la acción proviene de *introteams.php* implica que se ha de crear un nuevo agrupamiento y por tanto se borra la relación anterior que pudiera tener esta actividad con otro agrupamiento de la tabla '*blended_assign_grouping*' y se crea el nuevo agrupamiento, con el nombre recogido del formulario mediante *\$_POST*, con la función *groups_create_grouping()*.

Si la acción proviene de *updateTeams.php* implica que se desea borrar todos los grupos del agrupamiento llamando a la función *blended_delete_grouping()* y eliminando todas las calificaciones anteriores que pudieran existir de las tablas '*blended_grade*' y '*grade_grades*'.

Se mostrará mediante el formulario *teamsform* una tabla con los grupos con las casillas vacías si el método de creación es manual o rellenas con los códigos de identificación de los usuarios elegidos y sus fotos si la creación es aleatoria.

Por último, se muestra el botón '*Guardar Equipos*' que dirige el guardado de equipos al *script saveteams.php*.

Fichero saveteams.php

A este *script* se accede desde *createTeams.php* y posee los mismos permisos que éste.

Se recoge del formulario *teamsform* de *createTeams.php* el número de equipos *numteams* y se recorren mediante un bucle *for* para cada uno de ellos para obtener el nombre que posee guardándolo en *\$teamname* y los miembros, cuyo número se recoge en *\$nummembers*, se recorren con otro bucle *for* obteniendo los códigos de identificación que poseen las casillas del formulario *teamsform* que se rellanaron en el *script createTeams.php* y guardándolo en *\$insertmembers*.

Primero se comprueba que el profesor ha introducido alumnos, si no es así, se mostrará un mensaje de aviso y volverá a *teamsmanagement.php*.

Si se han introducido alumnos se guardarán todos los grupos mediante la función *blended_insert_teams()*.

Por otro lado, mediante la función *blended_num_members()* se recogerá en *\$contador* el número de repeticiones de cada alumno cuyo valor se pasará a *updateTeams.php* mediante el formulario *insertteams*.

Fichero grades.php

A este *script* se accede desde *view.php* y está encargado de mostrar una tabla con todas las actividades del curso que son calificables. Posee los permisos dados por el fichero *access.php* como:

```
'mod/blended:introteams' => array (
    'capytype' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_PREVENT,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

Como se puede observar, ni los alumnos ni los invitados pueden acceder a este *script*.

En esta página se muestra en una tabla con la función *html_writer::table(\$table)* el objeto *\$table* con todas las actividades del curso que son calificables. Para poder hacer esto, se recogen dichas actividades mediante *new grade_tree()*.

Con el uso de la función *get_coursemodules_in_course()* se obtiene la información de cada una de las actividades listadas como por ejemplo la fecha de entrega de la actividad.

En la tabla mostrada en la página aparecerá:

- La fecha de entrega de la actividad: el formato viene dado por *userdate()*. Si la actividad no posee esta característica el cuadro aparecerá vacío.
- Número de equipos definidos:
 - Si hay equipos definidos, se mostrará:
 - Número de equipos: obtenido de las tablas de los agrupamientos y guardado en la variable *\$numteams*.
 - Enlace a calificar equipos: este enlace dado por *\$gradelink* permite ir al *script introgrades.php* donde se pueden calificar e incluso modificar los equipos del agrupamiento que posee la tarea. Este enlace se puede presentar de tres maneras diferentes según el valor de *\$grades* que posee las entradas de la tabla *blended_grade* de la base de datos. De esta manera, si todos los grupos están calificados se mostrará 'Si'; si sólo están calificados algunos de los grupos entonces se mostrará 'Parcialmente'. En

estos dos casos el valor de *\$graded* será 1 indicando que ya se han introducido algunas calificaciones. Si no hay ninguno que se encuentre calificado entonces se mostrará 'No' y el valor de *\$graded* será 0.

- Si no hay equipos definidos
 - El cuadro de número de equipos se mostrará vacío.
 - Enlace a 'inicializar equipos' dado por *\$teamlink*. permite inicializar equipos con un enlace al *script introteams.php*.

Fichero introgrades.php

A este *script* se accede desde *grades.php*, desde *teamsmanagement.php* y desde *saveteamsgrades.php*. Posee los permisos dados por *access.php* como

```
'mod/blended:introgrades' => array (
    'captive' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_PREVENT,
        'teacher' => CAP_ALLOW,
        'editingteacher' => CAP_ALLOW,
        'manager' => CAP_ALLOW
    )
),
```

A este *script* no tienen permiso de acceder ni los alumnos ni los invitados, sólo profesores o administradores.

Este fichero está encargado de mostrar los equipos que posee el agrupamiento que se asignó a la actividad para poder calificarlos o incluso modificarlos si el profesor así lo desea.

Primero se comprueba que la actividad existe en la base de datos mediante la función *get_coursemodule_from_id()*. Si existe entonces se recoge su nombre y su identificador y se almacenan en *\$assignmentname* y *\$id_assignment* respectivamente.

Para mostrar los grupos se recoge de la base de datos los equipos y los miembros mediante las funciones *groups_get_all_groups()* y *groups_get_members()*. Por otro lado, mediante la función *blended_num_members()* se introduce en la variable *\$repetir_miembro* el número de veces que aparece cada miembro para comprobar si hay miembros que se encuentren repetidos en distintos grupos. Si esta situación se da, el

miembro que aparezca repetido se recuadrará en rojo indicando mediante un mensaje de aviso al profesor de la situación. Para que las tablas de los agrupamientos y de *blended* se encuentren sincronizadas se utiliza la función *blended_update_teams_by_groups()*.

Mediante el formulario *teamsform* se muestra una tabla con los grupos y sus miembros recogidos anteriormente. En este caso, además del nombre del equipo, los miembros y sus fotos, se mostrará un recuadro con la calificación introducida y una lista desplegable ('Calificación') con las calificaciones que se le pueden introducir a cada grupo obteniendo la calificación máxima que se le puede asignar a un equipo mediante la función *blended_get_grademax()*.

A la hora de introducir las calificaciones se actualizan dos tablas: *blended_grade* y *grade_grades*. Ambas tablas deben estar sincronizadas para lo cual se recogen mediante la función *blended_comprobar_calificacion()* ambas calificaciones pasando como parámetro final '*blended*' si se desea la calificación almacenada en *blended_grade* almacenándola en *\$grades_blended* o '*grades*' si se desea obtener la calificación de *grade_grades* almacenándola en *\$grades_grades*. Para cada entrada en cada una de estas variables mediante un bucle *foreach* se recogen cada una de las calificaciones en *\$grade_bl* y *\$grade_gr* respectivamente. Mediante estos dos *arrays* se comprobarán si poseen las mismas calificaciones o no. Si no son iguales, se mostrará al lado del usuario que las posea distintas un mensaje de aviso al profesor, indicándole que la calificación individual es distinta a la grupal y por tanto si guarda la grupal, la individual se perderá.

Si un equipo está vacío se mostrará un mensaje en rojo con '*Equipo vacío*' y si posee miembros pero no está calificado el mensaje que aparece es '*No calificado*'.

Cualquier cambio en los grupos se guardará y por tanto para la comodidad del profesor, se introduce un cuadro vacío en cada equipo por si desea introducir más miembros y un equipo vacío para introducir un nuevo grupo si así lo desea.

Por último se muestran dos botones al usuario:

1. Botón '*Ir a: Calificar Tareas*': este botón permite al usuario regresar a *grades.php*.
2. Botón '*Guardar calificaciones*': mediante este botón el usuario se dirige al *script* que permite guardar todos los cambios introducidos: *saveteamsgrades.php*

Fichero saveteamsgrades.php

A este *script* se accede desde *introgrades.php* y posee los mismos permisos que éste.

Al igual que en el *script* anterior, primero se comprueba con la función *get_coursemodule_from_id()* si la actividad existe en la base de datos. Si la actividad existe, entonces se recogen los parámetros *\$numteams* y *\$nummembers* del formulario *teamsform*. Mediante un bucle *for* se recorren los equipos introduciendo en *\$teamname* el nombre de cada equipo y en *\$id_team* su identificador. Mediante otro bucle *for* se recorren los miembros introduciendo en *\$memberid* su código de identificación. Con respecto a las notas introducidas, se guardan en *\$id_grade* el identificador de la calificación, *\$glvalue* el nombre de la lista desplegable 'calificación' y en *\$gvalue* la calificación elegida.

Para cada equipo se crea un objeto *\$team* con el identificador del equipo, el identificador de la actividad y el nombre del equipo. Y se crea un objeto *\$grade* con los parámetros: identificador, identificador del equipo, identificador de la actividad y calificación. El valor del último parámetro será distinto según se dé uno de los siguientes casos:

1. Si *\$id_grade* no es nulo y *\$gvalue* no es nulo: el valor de la calificación será *\$gvalue*.
2. Si *\$id_grade* no es nulo pero *\$gvalue* es nulo: el valor de la calificación será 0.
3. Si *\$id_grade* es nulo y *\$gvalue* no es nulo: la calificación tendrá el valor *\$gvalue* y el objeto *\$grade* no tendrá identificador.

Cada objeto *\$grade* creado para cada equipo se guardará en el array *\$updategrades*.

Mediante el uso de la función *blended_actualizar_agrupamiento()* se actualizarán los grupos y las calificaciones introducidas recogiendo en la variable *\$contador* el número de repeticiones de cada alumno. Tras actualizar las tablas de los agrupamientos, se deben actualizar las tablas de los grupos de *blended* mediante *blended_update_teams_by_groups()*. Mediante la función *blended_insert_teams()* se actualizará la tabla *blended_grade* y con las funciones *grade_update()* y *update_final_grade()* se actualizan las calificaciones de *grade_grades*.

Una vez realizadas las actualizaciones, se redirige la página al *script introgrades.php*.

Fichero selectassignment.php

A este *script* se accede desde *view.php* y está orientado al uso por los alumnos para poder inscribirse en un grupo de trabajo en alguna de las actividades creadas en el curso. Es por ello por lo que posee los siguientes permisos dados por *access.php* como

```

'mod/blended:signupteam' => array (
    'capytype' => 'read',
    'contextlevel' => CONTEXT_MODULE,
    'legacy' => array (
        'guest' => CAP_PREVENT,
        'student' => CAP_ALLOW,
        'teacher' => CAP_PREVENT,
        'editingteacher' => CAP_PREVENT,
        'manager' => CAP_PREVENT
    )
),

```

Se muestra mediante el formulario *signupteam* una lista desplegable con todas las actividades del curso que son calificables y que pueden poseer grupos de trabajo obtenidas mediante la función *grade_tree()* almacenándolas en la variable *\$items*. Mediante un bucle *foreach* se recorre cada actividad mostrando su nombre en la lista desplegable.

Tras la lista se muestra el botón '*Inscribirse*' pasando como parámetros la instancia y el nombre del módulo de la actividad seleccionada dirigiéndose a *signupteam.php*.

Fichero signupteam.php

A este *script* se accede desde el fichero *selectassignment.php* y desde *savesignupteam.php* poseyendo los mismos permisos de acceso que estos.

Si la acción proviene de *selectassignment.php* se recogen del formulario *signupteam* en *\$id_type_assignment* el identificador de la instancia de la actividad y en *\$tipoTarea* el nombre del módulo al que pertenece la actividad.

Primero se comprueba que la actividad existe y si es así, entonces se recoge su nombre en *\$assignmentname* y su identificador en *\$assignid*.

Se recogen en el *array* *\$teams* los equipos del agrupamiento que posee la actividad escogida. Si no posee equipos *\$teams* será un array vacío.

En las opciones del módulo el profesor puede elegir la creación de equipos de tres maneras distintas: Por el profesor, por los alumnos o por los alumnos con líder.

Primero se comprueba si existe un agrupamiento asignado a esa actividad y entonces se pueden dar las siguientes situaciones:

1. Los equipos los crean los profesores:

- No posee ningún agrupamiento asignado: se mostrará un mensaje al alumno de que no se puede inscribir en ningún equipo hasta que el profesor no le asigne un agrupamiento y vuelve al *script selectassignment.php*.
- Posee agrupamiento asignado: se le muestra al alumno el formulario *signupteamform* con una tabla con los grupos y los miembros que posee el agrupamiento.
 - o Si no está inscrito en ningún equipo se muestra la lista desplegable *'team'* con los nombres de los grupos para seleccionar el grupo en el que se desea inscribir y el botón *'Inscribirse en equipo'* que redirige a *savesignupteam.php* pasando el parámetro *\$action* con el valor *'signup'*.
 - o Si está inscrito en algún equipo sólo se mostrarán los equipos que posee el agrupamiento ya que no tiene la posibilidad de inscribirse en otro.

Si el agrupamiento no posee grupos, se le muestra un mensaje por el cual no puede crear grupos ya que no tiene permiso redirigiendo a *selectassignment.php*.

2. Los equipos los crean los alumnos:

- No posee ningún agrupamiento asignado: En este caso, el alumno tiene la posibilidad de crear un agrupamiento y un grupo en el que esté él. Para ello se mostrará un cuadro vacío (*name_team*) para escribir el nombre del equipo que desea introducir y el botón *'Nuevo equipo'* para crearlo. Al agrupamiento se le asignará el nombre de la actividad en la que es creado pasando la variable *\$action* con el valor *'new'* y la variable *\$nuevo* con el valor *'nuevo_agr'*.
- Tiene asignado un agrupamiento:
 - o Si está inscrito en un grupo:
 - Si no es líder: No podrá realizar ningún cambio y se mostrarán únicamente los grupos del agrupamiento.
 - Si es líder de un equipo: podrá:
 - Eliminar miembros de su equipo
 - Borrar el equipo entero
 - Eliminarse a sí mismo: en este caso el líder pasará a ser el siguiente alumno del grupo

En estos tres casos se pasará *\$action* como *'delete'*.
 - o Si no está inscrito en ningún grupo: podrá
 - Inscribirse en un equipo eligiendo mediante el desplegable el equipo en el que se desea inscribir y pasando *\$action* como *'signup'*.

- Crear un nuevo equipo: Mediante el cuadro *name_team* se introduce el nombre del equipo y con el botón '*Nuevo equipo*' se creará uno nuevo pasando *\$action* como '*new*' y *\$nuevo* como '*existente*'.

Cualquiera de las acciones mencionadas se mandarán al *script savesignupteam.php* para que sea realizada la acción escogida.

Fichero savesignupteam.php

A este *script* se accede desde *signupteam.php* y posee los mismos permisos de acceso que este.

Mediante este *script* se pueden realizar las siguientes acciones según el valor de *\$action* recogido del formulario *signupteamform*:

1. Inscribirse en un equipo: Si *\$action* posee el valor '*signup*' significa que un alumno desea inscribirse en un equipo. Primero se comprueba que el grupo no esté lleno ya que si es así se muestra un aviso al alumno de que el equipo está completo volviendo a *signupteam.php*. Si no está completo, se usará la función *blended_incribir_en_equipo_por_alumno()* donde primero se comprobará que no esté inscrito en ningún otro equipo.
2. Crear un equipo: Si *\$action* posee el valor '*new*' entonces el alumno desea crear un nuevo equipo. Se pueden dar dos casos distintos:
 - Crear un equipo dentro de un agrupamiento ya existente: se da si *\$nuevo* pose el valor '*existente*'. Se comprueba que el alumno tiene permiso para crear el grupo y se llama a la función encargada de crearlo: *blended_crear_equipo_por_alumno()* . En esta función se comprobará que no esté inscrito en ningún otro equipo y dependiendo de si el alumno puede ser líder o no se actualizarán las tablas *blended_team* y *blended_member* con la característica de líder activada o desactivada.
 - Crear un nuevo equipo y un nuevo agrupamiento: este caso se da si *\$nuevo* pose el valor '*nuevo_agr*'. Se comprueba que el tipo de creación de equipos activada permite a un alumno crear un nuevo grupo y un nuevo agrupamiento y si es así entonces se llama a la función *blended_crear_nuevo_agrupamiento_por_alumno()* encargada de crear el agrupamiento y el grupo y dependiendo de si el alumno puede ser líder o no se actualizarán las tablas *blended_team* y *blended_member* con la característica de líder activada o desactivada.

3. Borrar: Si *\$action* es *'delete'* y el tipo de creación de equipos en *'por alumnos con líder'* entonces el alumno podrá eliminar un miembro(o a sí mismo) o borrar el grupo entero. Para ello se hace uso de la función *blended_delete_sign_member_or_group()*. Si el líder se elimina a sí mismo se actualiza la tabla *blended_member* asignando al siguiente miembro del grupo la característica de líder.

Una vez realizados los cambios se redirige al *script signupteam.php*.

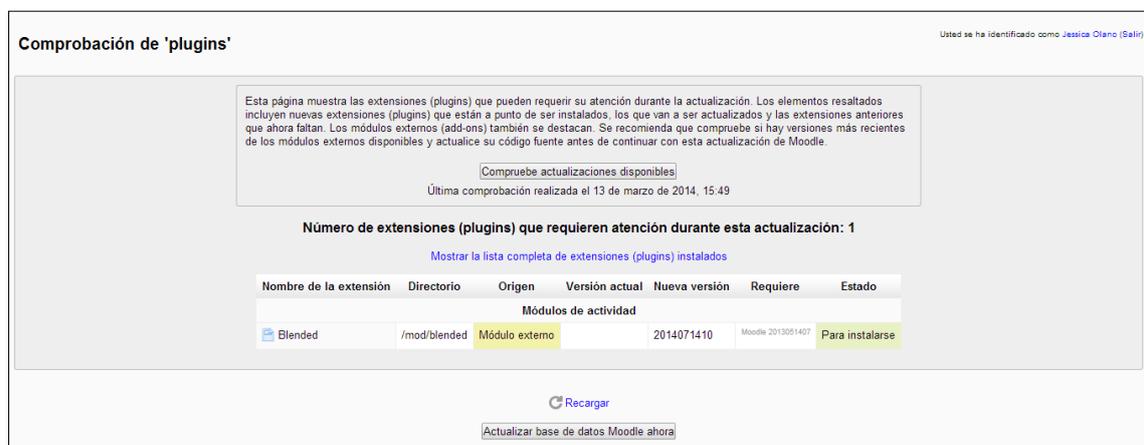
Capítulo 5. Análisis de resultados: manual de usuario

5.1. Instalación y desinstalación de *Blended*

5.1.1. Instalación

Para el uso del módulo *Blended* en la plataforma *Moodle* es necesaria su correcta instalación.

El primer paso es introducir la carpeta del módulo en `/moodle/mod` en cuya carpeta se encuentran todos los módulos que está utilizando *Moodle*. Tras esto, se debe entrar en *Moodle* como administrador y dirigirse a la página principal del servidor `http://localhost/moodle/admin/index.php`.



The screenshot shows the Moodle 'Comprobación de plugins' (Plugin Check) page. It displays a message about plugin updates and a table of plugins requiring attention. The 'Blended' module is listed as 'Módulo externo' with version 2014071410 and a 'Para instalarse' (To be installed) status.

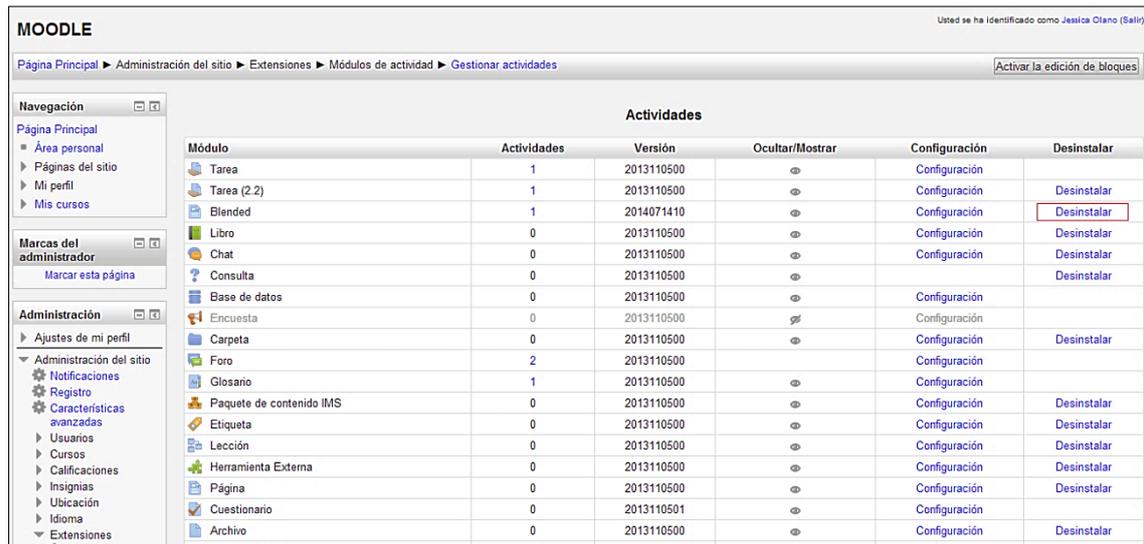
Nombre de la extensión	Directorio	Origen	Versión actual	Nueva versión	Requiere	Estado
Módulos de actividad						
Blended	/mod/blended	Módulo externo	2014071410	Moodle 2013051407		Para instalarse

Figura 1: Instalación de *Blended*

Si se pulsa el botón que permite la actualización de *Moodle* y por tanto en este caso la instalación de *Blended* se obtendrá, si todo ha ido correctamente, un mensaje que indica que la actualización ha tenido éxito. Con la instalación del módulo se crearán las tablas que éste posee en la base de datos de manera automática.

5.1.2. Desinstalación

Si se desea, el módulo puede ser eliminado. Para esto, se debe acceder a la plataforma como administrador y dirigirse a la página de actividades situada en la ruta <http://localhost/moodle/admin/modules.php>. Para eliminar el módulo basta con pulsar el botón desinstalar.



The screenshot shows the Moodle administrator interface. The main content area displays a table of activities. The 'Blended' module is highlighted with a red box around its 'Desinstalar' button.

Módulo	Actividades	Versión	Ocultar/Mostrar	Configuración	Desinstalar
Tarea	1	2013110500	☺	Configuración	
Tarea (2.2)	1	2014071410	☺	Configuración	Desinstalar
Blended	1	2014071410	☺	Configuración	Desinstalar
Libro	0	2013110500	☺	Configuración	Desinstalar
Chat	0	2013110500	☺	Configuración	Desinstalar
Consulta	0	2013110500	☺	Configuración	Desinstalar
Base de datos	0	2013110500	☺	Configuración	
Encuesta	0	2013110500	☺	Configuración	
Carpeta	0	2013110500	☺	Configuración	Desinstalar
Foro	2	2013110500	☺	Configuración	
Glosario	1	2013110500	☺	Configuración	
Paquete de contenido IMS	0	2013110500	☺	Configuración	Desinstalar
Etiqueta	0	2013110500	☺	Configuración	Desinstalar
Lección	0	2013110500	☺	Configuración	Desinstalar
Herramienta Externa	0	2013110500	☺	Configuración	Desinstalar
Página	0	2013110500	☺	Configuración	Desinstalar
Cuestionario	0	2013110501	☺	Configuración	
Archivo	0	2013110500	☺	Configuración	Desinstalar

Figura 2: Desinstalación de *Blended*

Al eliminar el módulo se eliminarán las tablas de las bases de datos que posea lo cual puede observarse accediendo a la base de datos *MySQL* existente en la dirección <http://localhost/phpmyadmin> y seleccionando la base de datos *Moodle* cuyas tablas poseen el prefijo *mdl_*.

5.2. Configuración de *Blended*

Una vez creada en el curso una instancia de *Blended* es posible modificar sus características para configurar el módulo según distintas opciones que aparecen en un formulario en la página *editar ajustes* del módulo.

Lo primero que aparece son los campos *Nombre de tarea* y *Descripción* que permite introducir el nombre que se desea que posea la instancia del módulo y un editor *HTML* que permite introducir una breve descripción de éste. Esta descripción puede ser mostrada si se desea en la página del curso activando la opción que lo permite.

Tras esto posee distintos desplegables con opciones para el módulo:

- Método de identificación:
 - o *Codificado*: identificador codificado mediante operación XOR con clave aleatoria.
 - o *Plano*: identificador sin codificar.

- Tipo de código de barras: tipo de código de barras mostrado a la hora de obtener las etiquetas de los alumnos:
 - o *QR2D*
 - o *C39*
 - o *EAN13*

- Número de columnas: selección del número de columnas en las que se dispondrán las preguntas en el cuestionario en formato PDF.

- Identificador: tipo de identificador de usuario a utilizar:
 - o *Número de identificación de Moodle*.
 - o *Id de usuario*: código de identificación del usuario
 - o *Id personalizado*: DNI o NIE

- Longitud del id personalizado: selección de la longitud del identificador personalizado.

- Creación de equipos: método de creación de equipos
 - o *Por el profesor*: los grupos de trabajo sólo pueden ser creados por el profesor.
 - o *Por los alumnos*: los equipos pueden ser creados por el profesor y por los alumnos pero estos últimos no los pueden modificar.
 - o *Por los alumnos con líder*: los alumnos pueden crear equipos y modificar los equipos creados por ellos.

- Tarea: tarea por defecto para la creación de equipos
 - o Cualquiera
 - o “Tareas que existan”

- Número de equipos por defecto: número máximo de equipos que una actividad puede poseer.

- Número de miembros por defecto: número máximo de miembros que un equipo puede tener.

Una vez realizados los cambios deseados, se pulsa el botón “*Guardar cambios y regresar al curso*” o “*Guardar cambios y mostrar*” si a demás de guardar, se desea mostrar la instancia del módulo.

Figura 3: Configuración de *Blended*

5.3. Manejo de *Blended*

Una vez configurado el módulo de la manera deseada, es posible la utilización de las herramientas que éste posee. En esta sección se muestra de manera detallada las herramientas que *Blended* posee, como acceder a ellas y las funciones que ofrecen cada una. Según el rol que el usuario posea, es decir, dependiendo de si el usuario es un profesor o un alumno, *Blended* poseerá distintas funcionalidades y por tanto se mostrarán distintas páginas para profesores que para alumnos.

5.3.1. Páginas disponibles tanto a profesores como alumnos

5.3.1.1. Página principal de *Blended*

La página principal de *Blended* es la página de una instancia creada del módulo donde aparecen todas las funcionalidades de las que consta. No permite el acceso a invitados y dependiendo del rol que posea se mostrarán distintos vínculos según las actividades que el usuario pueda realizar por el rol que posee.

-Vínculos visibles al profesor

Los vínculos configurados en el módulo y que posee el profesor son los siguientes:

Blended

Módulo Blended

¡ Hay estudiantes en el curso que no están activos !

¡ Hay estudiantes en el curso que no tienen asignado el código de identificación !

Generar página de etiquetas



[Generar página de etiquetas](#)

Imprimir páginas de pegatinas de identificación de los alumnos para pegar en los cuestionarios.

Podrá configurar el número de etiquetas que desea generar por página así como seleccionar los alumnos para los que desea generar las etiquetas.

Gestión de equipos y de tareas



[Generar página de tarea](#)

Esta sección le permitirá generar un documento para responder a una entrega de Moodle.





[Calificar tareas](#)

Introducir calificaciones para una tarea seleccionada. Permite calificar por equipos.



[Administración de equipos](#)

Permite la creación de equipos para cada tarea, de forma manual o aleatoria.

Figura 4: Vínculos visibles al profesor

- *Generar páginas de etiquetas:* El profesor tiene la opción de generar páginas de etiquetas con el tipo de etiqueta seleccionada en la configuración de *Blended* para los estudiantes matriculados en el curso que desee.
- *Generar página de tarea:* Con este enlace será posible la generación de páginas de tareas creadas en el curso para los estudiantes que el profesor desee.
- *Calificar tareas:* Una vez creados equipos para una actividad, el profesor tiene la posibilidad de calificarlos.
- *Administración de equipos:* Un profesor tiene la capacidad de asignar equipos a las actividades del curso que desee.

Como se puede observar aparecen dos mensajes. Estos mensajes son utilizados para alertar al profesor de situaciones que se están dando en el curso.

El primero de ellos se muestra cuando hay alumnos matriculados en el curso que no están activos. El segundo indica que hay estudiantes que no tienen asignado código de identificación.

Si se diera el caso de que ninguno de los estudiantes matriculados se encuentre activo, se mostrará el mensaje *¡No hay ningún estudiante activo en el curso!*.

-Vínculos visibles al alumno

Los vínculos que el alumno puede utilizar son los siguientes:

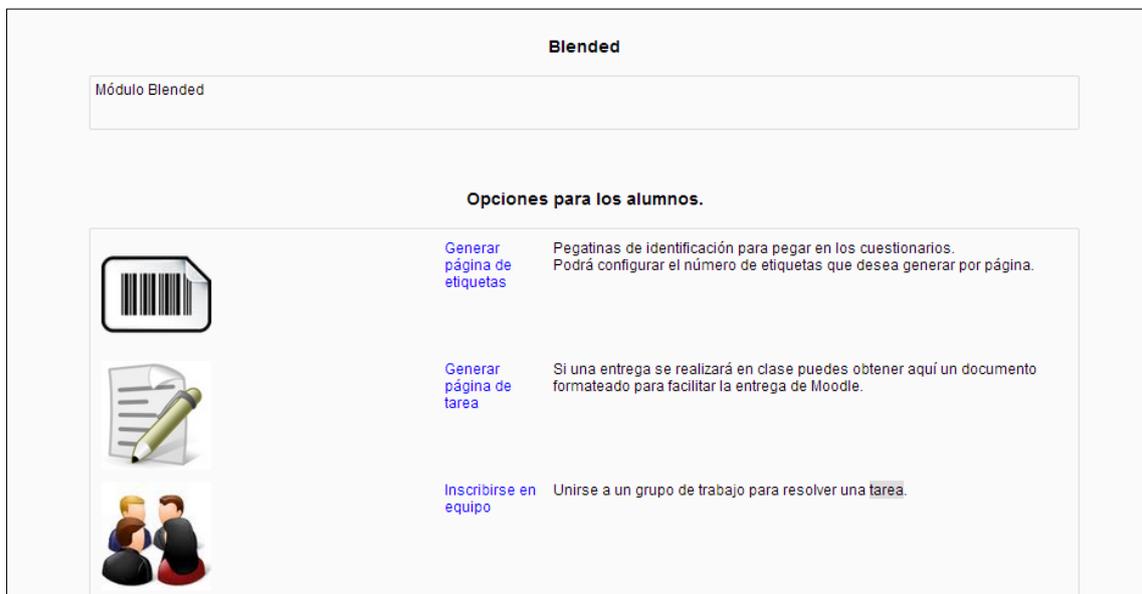


Figura 5: Vínculos visibles al alumno

- *Generar página de etiquetas*: El alumno tiene la posibilidad de obtener una página con las etiquetas correspondientes a su código de identificación.
- *Generar página de tarea*: Para cada una de las tareas creadas en el curso, el alumno tiene la posibilidad de imprimir su página de tarea con su código de identificación.
- *Inscribirse en equipo*: Un alumno tiene la posibilidad de inscribirse en un grupo de trabajo en cada actividad calificable creada en el curso.

5.3.1.2. Generar página de etiquetas

A esta página se accede desde el vínculo *Generar página de etiquetas* mediante la cual se pueden obtener páginas rellenas con códigos asociados a los estudiantes. Estos códigos pueden ser de tipo QR2D, C39 o EAN13. Mediante estas etiquetas se identifica de manera unívoca a cada estudiante y pueden ser utilizadas tanto en trabajos como en exámenes así como cualquier otra actividad planteada por el profesor.

Tanto si el usuario posee el rol de estudiante como de profesor, se le muestra una tabla con distintas opciones que permiten definir el formato de la página que se va a mostrar:

- Formato de página: desplegable con el tipo de folio a utilizar
- Márgenes superior e inferior tanto derecho como izquierdo
- Número de filas a imprimir
- Número de columnas a imprimir

En la definición del formato de las etiquetas el usuario puede elegir entre tres opciones:

- No identificar: en la etiqueta sólo aparecerá el código
- Mostrar código ID legible: además del código se muestra el código de identificación que el alumno tiene asignado.
- Mostrar nombre completo del usuario: además del código se muestra el nombre completo del alumno.

Si el usuario posee el rol de alumno la página a mostrar contendrá las opciones descritas.

Generar página de etiquetas

Seleccione el número de etiquetas por fila y por columna que desee imprimir

Formato de Página:

Formato de Página	A4 ▼	Margen superior mm. (Mídalo cuidadosamente en su hoja de pegatinas)	5	Margen inferior mm. (Mídalo cuidadosamente en su hoja de pegatinas)	5
		Margen izquierdo mm. (Mídalo cuidadosamente en su hoja de pegatinas).	8	Margen derecho mm. (Mídalo cuidadosamente en su hoja de pegatinas).	8

Número de filas: 17 ▼

Número de columnas: 4 ▼

Formato de las etiquetas:

No identificar
 Mostrar código ID legible
 Mostrar nombre completo del usuario

[Imprimir etiquetas](#)

Figura 6: Generar página de etiquetas por alumno

Si el usuario posee el rol de profesor, además de estas opciones se le mostrará un cuadro con los alumnos matriculados en el curso para que pueda elegir los alumnos de los que desea imprimir sus etiquetas. Por otro lado, también tiene la posibilidad de elegir el método de disposición de las etiquetas:

- Una etiqueta por cada estudiante activo en el curso
- Una etiqueta por cada estudiante matriculado en el curso
- Páginas completas para todos los estudiantes matriculados que aparecen en la lista mostrada que sean elegidos por el profesor.

Generar página de etiquetas

Seleccione el alumno y número de etiquetas por fila y por columna que desee imprimir

Formato de Página:

Formato de Página	A4	Margen superior mm. (Mídalo cuidadosamente en su hoja de pegatinas)	5	Margen inferior mm. (Mídalo cuidadosamente en su hoja de pegatinas)	5
		Margen izquierdo mm. (Mídalo cuidadosamente en su hoja de pegatinas).	8	Margen derecho mm. (Mídalo cuidadosamente en su hoja de pegatinas).	8

Número de filas: 17

Número de columnas: 4

Imprimir para un alumno:

Alumno: Estudiante 3
Estudiante 4
Estudiante 5
Estudiante 6
Estudiante 7 (*) (**)

¡ Con (*) los estudiantes no activos en el curso !

¡ Con (**) los estudiantes en el curso que no tienen asignado el código de identificación !

Métodos de disposición de las etiquetas:

Una etiqueta por cada estudiante activo en el curso.

Una etiqueta por cada estudiante matriculado en el curso.

Páginas completas de etiquetas para los estudiantes seleccionados en la lista de la izquierda.

Formato de las etiquetas:

No identificar

Mostrar código ID legible

Mostrar nombre completo del usuario

Figura 7: Generar página de etiquetas por profesor

Puede darse el caso de que haya alumnos en el curso que no estén activos o que no posean código de identificación. En estos casos, se muestran al lado de los alumnos que poseen estas características los caracteres (*) y/o (**) indicando que no están activos y/o que no tienen asignado código de identificación respectivamente. En caso de que estos caracteres aparezcan, se muestran en rojo los mensajes correspondientes para alertar al profesor de la situación.

Para ambos tipos de usuario, al pulsar el botón *Imprimir etiquetas* se mostrará la página con las etiquetas con el formato y las opciones escogidas por el usuario. Como ejemplo, se muestra a continuación una página de etiquetas para el “*estudiante 3*” con los parámetros escogidos por defecto y mostrando el

nombre completo, realizado con la opción de identificador de usuario escogida y con dos tipos de códigos:

- Si el código escogido en la configuración de *Blended* es EAN13: se muestran códigos de barras con el código de identificación del alumno



Figura 8: Etiquetas EAN13

- Si el código escogido es QR2D:

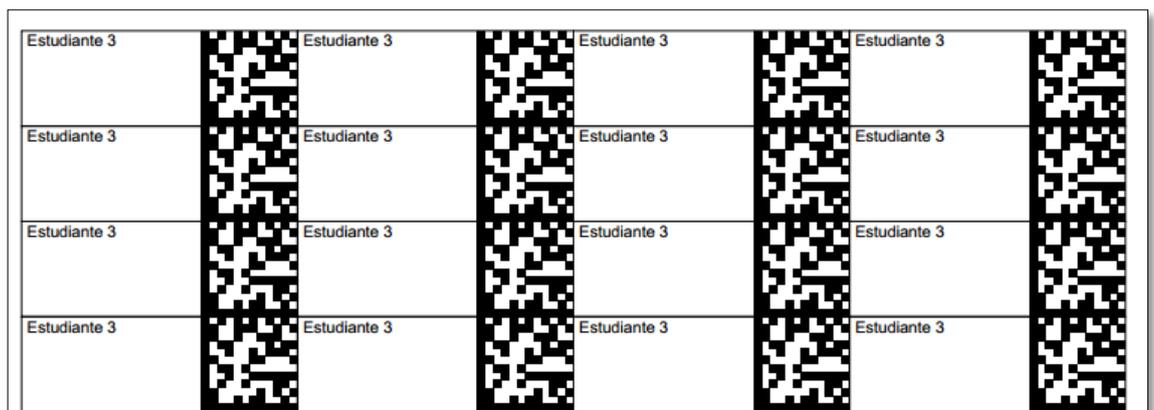


Figura 9: Etiquetas QR2D

Si el estudiante no posee código de identificación no se podrá imprimir sus etiquetas, por ello se mostrará un mensaje al usuario indicándole la situación:



Figura 10: Mensaje de alerta generación de etiquetas

Si se da al botón continuar vuelve a la página *Generar página de etiquetas*.

5.3.1.3. Generar página de tarea

A ésta página se accede desde el enlace *Generar página de tarea* y es utilizada para poder imprimir una página para una de las tareas creadas en el curso para un alumno.

Para todo usuario ya posea el rol de alumno o de profesor se muestra una lista desplegable con las tareas creadas en curso. Se obtendrá la página de la tarea elegida.

Si el usuario posee rol de alumno, este desplegable será lo único que poseerá:



Figura 11: Generar página de tarea por alumno

Si el usuario posee el rol de profesor, se le mostrará también un cuadro con la lista de alumnos matriculados en el curso. Se imprimirá la página de tarea para el estudiante seleccionado:

Generar página de tarea

Seleccione la hoja de **tarea** que desee imprimir

Selecciona la tarea: tarea

Alumno:

Estudiante 1
 Estudiante 2
 Estudiante 3
 Estudiante 4
 Estudiante 5

¡ Con (*) los estudiantes no activos en el curso !

¡ Con (**) los estudiantes en el curso que no tienen asignado el código de identificación !

Figura 12: Generar página de tarea por profesor

Si se da el caso de que haya alumnos en el curso que no estén activos o que no posean código de identificación. En estos casos, se muestran al lado de los alumnos que poseen estas características los caracteres (*) y/o (**) indicando que no están activos y/o que no tienen asignado código de identificación respectivamente. En caso de que estos caracteres aparezcan, se muestran en rojo los mensajes correspondientes para alertar al profesor de la situación.

Una vez elegidas las opciones, si se da al botón *Imprimir página de tarea* se obtendrá la página de la tarea elegida con un cuadro al inicio de la página con:

- Logo oficial de la universidad de Valladolid
- Nombre del curso
- Nombre de la tarea elegida
- Fecha límite de entrega de la tarea
- Nombre del alumno
- Etiqueta del alumno
- Descripción de la tarea

A modo de ejemplo se va a mostrar la página para la tarea con nombre “Tarea” para “estudiante 1” con el uso del tipo de código EAN13:

Nombre del curso:Curso 1 0 1 2 3 4 5 6 7 8 9 10

Nombre de tarea:tarea

Fecha límite de entrega:jueves 2, julio, 2015

Nombre del alumno:Estudiante 1

Universidad de Valladolid

Descripción de la [tarea](#)

Figura 13: Ejemplo página de tarea

Si se da el caso de que el estudiante no posea código de identificación entonces no se podrá imprimir la página de tarea mostrando un mensaje de la situación



Figura 14: Mensaje de alerta generación de página de tarea

Si se da al botón *Continuar* vuelve a la página *Generar página de tarea*.

5.3.2. Páginas disponibles únicamente a profesores

5.3.2.1. Administración de equipos

A esta página sólo pueden acceder aquellos usuarios cuyo rol sea de profesor y/o administrador. Se accede desde la página principal de *Blended* y tiene la funcionalidad de la creación y asignación de equipos en las actividades calificables creadas en el curso.

Administración de equipos						
Permite crear equipos para cada una de las tareas de forma manual o aleatoria						
Nombre de tarea	Fecha limite de entrega	Número de equipos	Calificada			
tarea	jueves 2, julio, 2015	2	No	Inicializar equipos	Modificar equipos	
Tarea_vieja	martes 7, julio, 2015	0		Inicializar equipos		
Foro		0		Inicializar equipos		
Glosario		0		Inicializar equipos		

Figura 15: Administración de equipos

La página principal de administración de equipos consta de una tabla donde se muestran las actividades, su fecha de entrega (en el caso de las tareas), el número de equipos que posee, si está o no calificada (“No”, ”Si” o

”Parcialmente”) que es un enlace que permite ir a calificar la actividad, y dos tipos de enlaces:

- *Inicializar equipos*: este enlace permite inicializar equipos para una actividad cuando no posee ningún agrupamiento asignado o cuando ya posee un agrupamiento con sus respectivos grupos pero se desea asignarle otro diferente.
- *Modificar equipos*: este enlace sólo se muestra cuando la actividad ya posee equipos y permite ir a la página donde los miembros de estos equipos pueden ser modificados.

A continuación se muestran el uso de los dos enlaces descritos:

1. *Inicializar equipos*: cuando se desea que una actividad posea los equipos de un agrupamiento ya sea porque aún no tiene asignado ninguno o porque lo tiene pero el profesor lo desea cambiar por otro se utiliza este enlace.

Gestión de equipos para Tarea_vieja

Eliga agrupamiento existente:
 ▼

Cree un nuevo agrupamiento:

Método de creación de equipos:	<input type="text" value="Manual"/> ▼
Selección de estudiantes:	<input type="text" value="Solo activos"/> ▼
Número de equipos:	<input type="text" value="10"/> ▼
Número de miembros por equipo:	<input type="text" value="4"/> ▼
Nombre Agrupamiento:	<input type="text"/>

Figura 16: Inicializar equipos

La asignación de un agrupamiento se puede realizar de dos maneras distintas:

- *Asignando un agrupamiento existente*: se muestra una lista desplegable con los agrupamientos guardados en la base de datos que poseen sus propios equipos ya creados. Si se selecciona uno de ellos y se pulsa el botón “continuar” se asigna automáticamente ese agrupamiento y sus grupos a esa actividad.

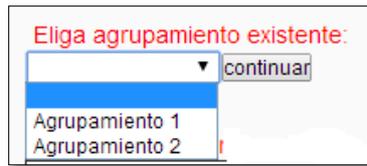


Figura 17: Elegir agrupamiento

- *Creando un nuevo agrupamiento:* si el profesor lo desea puede crear un nuevo agrupamiento con los equipos que él desee introduciéndolos de manera manual. Para ello se muestra un cuadro donde se puede elegir:
 - *Método de creación de equipos:* puede elegir entre manual o aleatorio. Si se elige aleatorio se rellenarán automáticamente los equipos con los alumnos que el profesor escoja (activos o todos los matriculados) elegidos con un desplegable.
 - *Número de equipos*
 - *Número de miembros por equipo*
 - *Nombre del agrupamiento*

Una vez introducidos los campos se pulsa el botón “*Nuevo agrupamiento*” mediante el cual se accede a la página que permite ver los equipos e introducir los miembros.

A modo de ejemplo se muestra a continuación la creación de un nuevo agrupamiento eligiendo el método manual con tres equipos de dos miembros cada uno:

Gestión de equipos para Tarea_vieja

Equipo:	1	Identificadores:	<input type="text"/>	<input type="text"/>
Equipo:	2	Identificadores:	<input type="text"/>	<input type="text"/>
Equipo:	3	Identificadores:	<input type="text"/>	<input type="text"/>

Figura 18: Nuevo agrupamiento

El nombre de los equipos se muestra de manera inicial rellenado mediante un contador que va desde 1 hasta el número de equipos. En este caso, el profesor está encargado de rellenar los grupos con los

códigos de identificación de los alumnos que desee introducir. Una vez rellenos se pulsa el botón “*Guardar equipos*” con el que se guardan los equipos en el agrupamiento creado, y por tanto la actividad ya tendrá su agrupamiento con sus equipos correspondientes.

2. *Modificar equipos*: Mediante este enlace se accede a la página que permite modificar los alumnos de los equipos del agrupamiento.

Gestión de equipos para tarea

Método de creación de equipos:

Selección de estudiantes:

Número de equipos:

Número de miembros por equipo:

[Reinicializar grupos](#)

Equipos de: Agrupamiento 2

Equipo	Identificadores	Estudiante 1	Estudiante 2	Estudiante 3	Estudiante 4	Estudiante 5	Estudiante 6
<input type="text" value="1"/>	<input type="text" value="11111111"/>		<input type="text" value="22222222"/>		<input type="text" value="33333333"/>		<input type="text" value="44444444"/>
<input type="text" value="2"/>	<input type="text" value="22222222"/>		<input type="text" value="55555555"/>		<input type="text" value="66666666"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

[Actualizar Agrupamiento](#) [Ir a: Administración de Equipos](#)

Figura 19: Modificar equipos

En esta página aparecen en una tabla el nombre del agrupamiento que tiene asignado la actividad junto con los equipos y sus miembros. Existen dos maneras distintas de modificar los grupos:

- *Modificar de manera manual los equipos*: para pequeños cambios, el profesor tiene la capacidad de modificar los alumnos que desee borrándolos, cambiándolos de grupo o añadiendo nuevos alumnos. Para hacer más sencillo los cambios, se deja libre un cuadro por cada equipo y un equipo vacío para que el profesor pueda añadir más miembros u otro equipo si así lo desea. Si se pulsa el botón “*Actualizar agrupamiento*” se guardarán los cambios mostrando de nuevo esta misma página con los nuevos equipos guardados.

Cuando el profesor introduzca un alumno que no exista en la base de datos, ese alumno no será guardado. Por otro lado, si introduce un mismo alumno en distintos grupos aparecerá ese alumno recuadrado en rojo y si

se coloca el cursor encima del recuadro aparecerá un mensaje indicando que el identificador del alumno se encuentra repetido.

Figura 20: Equipos con identificador repetido

- *Reinicializar los equipos del agrupamiento:* si en vez de pequeños cambios el profesor desea volver a inicializar los grupos del agrupamiento se presenta un cuadro donde puede elegir: el método de creación, seleccionar el tipo de estudiante si la creación es aleatoria, el número de equipo y el número de miembros por equipo. Si se pulsa el botón “*Reinicializar equipos*” se mostrará la página que permite introducir los miembros en los equipos poseyendo la misma vista que la obtenida cuando se crea un nuevo agrupamiento. Tras esto se volverá a esta página donde se mostrarán los nuevos grupos creados.

Para que al profesor le resulte cómodo se ha introducido el botón “*Ir a: Administración de equipos*” que permite volver a la página de administración de equipos donde se muestra la tabla con las actividades del curso.

5.3.2.2. Calificar tareas

A esta página se accede desde la página principal de *Blended* cuyo acceso está restringido a los profesores y/o administradores. Mediante esta página es posible calificar las actividades creadas en el curso.

Calificar tareas			
Tareas a calificar:			
Nombre de tarea	Fecha limite de entrega	Número de equipos:	Calificada
 tarea	jueves 2, julio, 2015	2	Parcialmente
 Tarea_vieja	martes 7, julio, 2015	1	No
 Foro		Inicializar equipos	
 Glosario		Inicializar equipos	

Figura 21: Calificar tareas

Se muestra una tabla con las actividades creadas en el curso que son calificables y por tanto pueden poseer grupos de trabajo, la fecha de entrega (en el caso de las tareas), el número de equipos que posee y si está o no calificada. En el caso de que no se haya definido un agrupamiento a una actividad, aparece el enlace a “*Inicializar equipos*” que lleva a la página proporcionada por este enlace ya explicada en el apartado anterior. Por otro lado, la indicación de si la actividad está o no calificada se puede presentar de tres formas diferentes:

- Si la actividad está calificada se muestra “*Si*”
- Si la actividad está calificada de manera parcial, es decir, no todos los grupos están calificados se muestra “*Parcialmente*”
- Si la actividad no posee calificación ninguna se muestra “*No*”

Los tres son enlaces que permiten ir a calificar la actividad. La página que califica la actividad posee la siguiente estructura:

Introducir calificaciones: tarea					
Equipos de: Agrupamiento 2					
Equipo: <input type="text" value="1"/>	Identificadores:	 <input type="text" value="33333333"/> Estudiante 3	 <input type="text" value="44444444"/> Estudiante 4	<input type="text"/>	Calificación: <input type="text" value="7"/> <input type="text" value="7"/>
Equipo: <input type="text" value="2"/>	Identificadores:	 <input type="text" value="11111111"/> Estudiante 1	 <input type="text" value="22222222"/> Estudiante 2	 <input type="text" value="66666666"/> Estudiante 6	Calificación: <input type="text"/> <input type="text" value="-"/> No calificado
Equipo: <input type="text"/>	Identificadores:	<input type="text"/>	<input type="text"/>	<input type="text"/>	Calificación: <input type="text"/> <input type="text" value="-"/> Equipo vacío
<input type="button" value="Guardar calificaciones"/> <input type="button" value="Ir a: Calificar Tareas"/>					

Figura 22: Introducir calificaciones

- El nombre de la actividad
- El nombre del agrupamiento

- Una tabla en la que para cada equipo se muestran sus miembros, se deja un espacio para introducir un miembro más si el profesor así lo desea y la calificación que posee el equipo. Si el equipo no está calificado se muestra el mensaje en rojo “*No calificado*”. Por otro último, se introduce un equipo vacío con el mensaje en rojo de “*Equipo vacío*” que permite al profesor introducir un nuevo grupo al agrupamiento.

Cuando se introducen cambios en los grupos ya sea en los miembros o en la calificación se actualizan tanto los agrupamientos como las tablas de calificaciones pertinentes al pulsar el botón “*Guardar calificaciones*”.

Si se da la situación en la que un alumno posee una calificación individual distinta a la grupal aparecerá al lado del alumno un símbolo de alerta que colocando el cursor sobre él muestra un mensaje avisando al profesor de que ese alumno posee una calificación individual distinta a la grupal y por tanto si guarda la grupal la individual desaparecerá sustituida por la grupal y mostrando cual es la nota individual que el alumno posee.

The screenshot shows a web interface titled "Introducir calificaciones: tarea". It displays a table for "Equipos de Agrupamiento 2".

Equipo	Identificadores	Calificación	Estado
1	33333333 (Estudiante 3), 44444444 (Estudiante 4)	7	
2	La calificación individual es distinta a la grupal, si cambia la grupal sobrescribirá la individual. Nota individual: 5		No calificado
			Equipo vacío

Buttons at the bottom: "Guardar calificaciones" and "Ir a: Calificar Tareas".

Figura 23: Alerta de calificación individual

Por último si el profesor lo desea puede volver a la página de calificar tareas mediante el botón “*Ir a: Calificar tareas*”.

5.3.3. Páginas disponibles únicamente a alumnos

5.3.3.1. Inscribirse en equipo

A esta página puede acceder el alumno desde la página principal de *Blended* de la que dispone y es usada por el alumno para poder inscribirse en un equipo de trabajo para una de las actividades calificables creadas en el curso. Al entrar

en la página se le presenta al estudiante una lista desplegable con dichas actividades para que elija aquella en la que se desea inscribir.

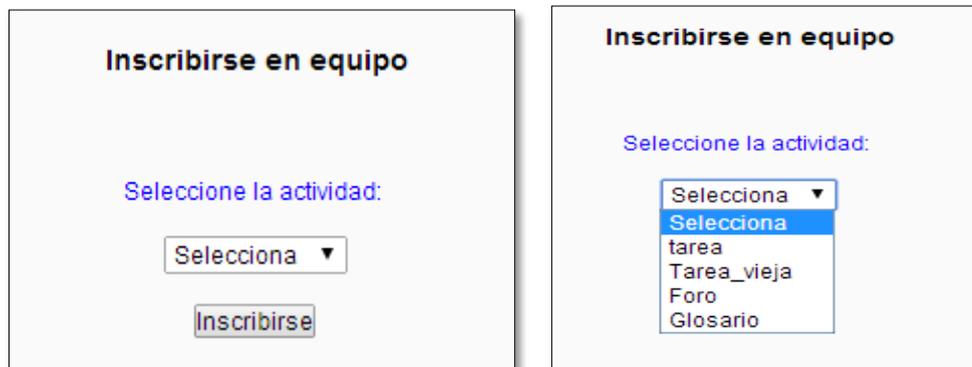


Figura 24: Inscribirse en equipo

Una vez elegida la actividad que desee, se pulsa el botón “*Inscribirse*” para poder manejar la inscripción dentro de la actividad seleccionada.

Dentro de las opciones que el administrador/profesor elija que tenga *Blended*, existe la opción “*Creación de equipos*” que permite elegir qué tipo de usuario tiene el permiso de crear nuevos equipos. Según el valor escogido el alumno puede obtener distintos mensajes a la hora de inscribirse en una actividad.

1. *Por los profesores*

Si los equipos de trabajo sólo los pueden realizar los profesores, entonces los alumnos sólo tendrán la capacidad de inscribirse en equipos ya formados y no podrán crear nuevos equipos. Por otro lado, es posible que se den las siguientes situaciones:

- *La actividad no tiene asignado un agrupamiento:*
Se le mostrará un mensaje por pantalla al alumno indicándole que hasta que no tenga asignado un agrupamiento no podrá inscribirse.
Mediante el botón “*Continuar*” vuelve a la selección de la actividad.



Figura 25: Mensaje de denegación de la inscripción

- *El alumno ya está inscrito en un equipo en la actividad:*
 En el caso en el que la actividad si posea agrupamiento y por tanto los grupos que éste contiene, si el alumno ya está inscrito se le mostrará una tabla con los equipos: nombre, los miembros que estos poseen y el número de miembros de cada equipo. No tiene la posibilidad de cambiarse de equipo.
 A modo de ejemplo, el estudiante 1 ya estaba inscrito en *Tarea* y por tanto no se puede volver a inscribir ni cambiarse de grupo:

Equipos	Miembros de equipo	Número de miembros
1	 Estudiante 3  Estudiante 4	2
2	 Estudiante 1  Estudiante 2  Estudiante 6	3

Figura 26: Alumno ya inscrito en un equipo

- *El alumno no está inscrito en ningún equipo en la actividad:*
 En el caso en el que la actividad si posea agrupamiento y por tanto los grupos que éste contiene, si el alumno no está inscrito en ninguno de ellos se mostrará una tabla con los grupos que el agrupamiento posee: nombres, los miembros de cada grupo y el número de miembros que posee. Tras esto se le mostrará al usuario la posibilidad de inscribirse en

alguno de los equipos existentes. Para ello, tras la tabla se muestra una lista desplegable con los nombres de los equipos y el botón “*Inscribirse en equipo*” mediante el cual el alumno será inscrito en el equipo elegido mostrándose de nuevo la página de la actividad con la tabla con los grupos y sin la posibilidad de cambiarse de grupo y/o inscribirse en otro.

A modo de ejemplo, se puede observar que en “*Tarea*” el alumno llamado “*Estudiante 5*” no se encuentra en ningún equipo, por tanto si este alumno se quiere inscribir en esta tarea aparecerá la siguiente página:

Inscribirse en equipo

Equipos	Miembros de equipo	Número de miembros
1	 Estudiante 3  Estudiante 4	2
2	 Estudiante 1  Estudiante 2  Estudiante 6	3

Selecciona ▾
 Selecciona
 1
 2

?

Figura 27: Ejemplo inscribirse en un equipo

Al no estar inscrito en ningún equipo se le muestra la posibilidad de inscribirse en uno de los ya creados apareciendo en el desplegable el nombre de ellos. Si por ejemplo selecciona el equipo “1” y presiona el botón inscribirse el alumno será inscrito en ese equipo y vuelve a la página de la actividad mostrando la tabla y apareciendo en ella sin la posibilidad de inscribirse en otro.

Inscribirse en equipo

Equipos	Miembros de equipo	Número de miembros
1	 Estudiante 3	3
	 Estudiante 4	
	 Estudiante 5	
2	 Estudiante 1	3
	 Estudiante 2	
	 Estudiante 6	

Figura 28: Ejemplo estudiante inscrito

2. Por los alumnos

Si los equipos de trabajo pueden ser realizados por los alumnos entonces no sólo tendrán la posibilidad de inscribirse en equipos ya formados, también podrán crear sus propios equipos. Se pueden dar las siguientes situaciones:

- *La actividad no tenga asignado ningún agrupamiento:*

Si la actividad no posee agrupamiento pero el alumno tiene la posibilidad de crear equipos, al introducirse en esa actividad, el alumno podrá crear un grupo con el que se creará de manera automática un agrupamiento con el mismo nombre de la actividad introduciendo el grupo creado por el alumno. Para ello se muestra un cuadro vacío donde el alumno introducirá el nombre del equipo que desea y el botón “*Crear nuevo equipo*”.

Inscribirse en equipo

Nombre de equipo (opcional)

Figura 29: Creación de agrupamiento por alumno

Si el nombre no es introducido se rellenará con un contador que en este caso al no haber ningún equipo tendrá el valor "1". Con esto el agrupamiento será creado y se mostrará la tabla en la que aparece su equipo con él como miembro y sin ninguna posibilidad de inscribirse o crear otro equipo.

Como ejemplo se muestra que el "Estudiante 5" crea un equipo en la actividad llamada "Foro" con el nombre "Equipo1":

Inscribirse en equipo		
Equipos	Miembros de equipo	Número de miembros
Equipo1	 Estudiante 5	1

Figura 30: Agrupamiento y grupo creado por alumno

- *Si el alumno ya está inscrito en un equipo:*
Si la actividad ya posee un agrupamiento y el alumno ya está inscrito en un equipo se le mostrará la tabla con los equipos y sus miembros de la misma manera que en el caso de creación de equipos por profesores.
- *Si el alumno no está inscrito en ningún equipo:*
Si la actividad ya tiene asignado un agrupamiento y el alumno no se encuentra inscrito en ningún grupo, entonces se le mostrará la tabla con los grupos: nombre, miembros y número de miembros. Tras la tabla tendrá dos posibilidades para inscribirse en la actividad:
 - o Inscribirse en un equipo existente: se le muestra la lista desplegable con el nombre de los equipos y el botón "Inscribirse en equipo". Si elige esta opción se inscribirá en el equipo elegido de la misma forma que si el método de creación elegido es por los profesores.
 - o Crear un nuevo equipo: al poder los alumnos crear equipos, tendrá la posibilidad de crear un nuevo equipo dentro del agrupamiento. Para ello, se muestra un cuadro en blanco donde introducir el nombre del equipo. Si el cuadro de deja vacío se rellena con un contador que le asignará el primer número que esté libre. Mediante el botón "Crear nuevo equipo" se creará el nuevo equipo dentro del agrupamiento con él como miembro.

A modo de ejemplo, la actividad “*Tarea_vieja*” posee un único grupo con el alumno “*Estudiante 6*” y por tanto el “*Estudiante 5*” tendrá las posibilidades descritas:

Inscribirse en equipo

Equipos	Miembros de equipo	Número de miembros
2	Estudiante 6	1

Figura 31: El alumno puede crear un equipo

Si introduce un nombre y crea un equipo se mostrará la tabla, en este caso, con el equipo con nombre 2 con “*Estudiante 6*” y el que él ha creado con él como alumno.

Inscribirse en equipo

Equipos	Miembros de equipo	Número de miembros
2	Estudiante 6	1
Equipo 3	Estudiante 5	1

Figura 32: Alumno crea un nuevo equipo

Una vez que el alumno esté en un equipo no podrá inscribirse y/o crear otro equipo

3. *Por los alumnos con líder*

Al igual que en el caso anterior, los alumnos tienen la posibilidad de crear nuevos grupos, pero con la diferencia de que el alumno que cree un grupo será líder de éste. Ser líder implica la posibilidad de modificar el grupo: eliminar miembros, eliminar el grupo entero o eliminarse a sí mismo.

- *Si la actividad no posee agrupamiento:*
Se mostrará la misma página que en el caso anterior creándose el agrupamiento de la misma manera. En este caso, cuando el grupo es creado el alumno será líder de éste y por tanto cuando la tabla con el grupo sea mostrada se verá en color azul negrita el nombre del alumno con la palabra líder al lado para informar al usuario de que es líder de equipo.
- *Si el alumno ya está inscrito en un equipo:*
Si la actividad ya posee un agrupamiento y el alumno ya está inscrito en un equipo se muestra la misma página que en los casos anteriores.
- *Si el alumno no está inscrito en ningún equipo:*
Si la actividad posee un agrupamiento y el alumno no se encuentra inscrito en ningún equipo entonces tendrá la posibilidad de inscribirse en uno o de crear uno nuevo. Se muestra la misma página que en el caso anterior con la diferencia de que si el alumno crea un nuevo equipo, cuando se muestre la tabla aparecerá su nombre en color azul negrita con la palabra líder a lado.

Cuando un alumno es líder de un equipo y entra en esa tarea donde es líder se le muestra su equipo con los alumnos que posea (porque por ejemplo se hayan inscrito en él) en color azul negrita y todos los alumnos incluido el líder tendrán un cuadro (“checkbox”) mediante el cual será posible seleccionar el/los alumnos que el líder desea eliminar.

A modo de ejemplo se muestra a continuación un equipo creado en “Tarea_vieja” por “Estudiante 1” que será el líder y donde posteriormente se ha inscrito el “Estudiante 2”. Cuando el líder entre se mostrará la siguiente página:

Inscribirse en equipo

Equipos	Miembros de equipo	Número de miembros
2	 Estudiante 6	1
Equipo 3	 Estudiante 5	1
Equipo4	<input type="checkbox"/> Estudiante 1 (Lider) <input type="checkbox"/> Estudiante 2	2

Figura 33: Equipo con líder

Mediante los *checkboxs* el alumno podrá seleccionar que alumnos desea eliminar mostrándole el botón “*Eliminar miembros*” con el que se realizará la acción.

Si el líder se borrara a sí mismo se nombrará líder de manera automática al siguiente alumno del equipo.

5.4. Copia de seguridad

Todo módulo en *Moodle* tiene la posibilidad de realizar una copia de seguridad de los datos almacenados en sus tablas en las bases de datos. Mediante las copias de seguridad es posible la realización de una restauración por ejemplo en el caso de que el sistema fallara. Para ello, los cursos se pueden empaquetar en un archivo *zip* usando la función “*copia de seguridad*” presente en la administración del curso.



Figura 34: Acceso a copia de seguridad

Parte 3. Conclusiones

Capítulo 6. Conclusiones

Hoy en día la tecnología es parte fundamental en el desarrollo académico. Cada vez es más significativo el uso de plataformas que permitan introducir de manera sencilla un método de aprendizaje que pueda ser útil en el ámbito académico.

Este trabajo forma parte de un proyecto de Innovación Educativa de la Universidad de Valladolid enfocado al desarrollo de un módulo en la plataforma de aprendizaje *Moodle* cuyo objetivo principal está enfocado a la complementación de la formación académica de los estudiantes y el desarrollo de herramientas que permitan a los profesores manejar los cursos de manera sencilla y cómoda.

Para el desarrollo de esta herramienta es necesario el conocimiento de los lenguajes *Javascript*, *PHP*, *HTML*, y el acceso y manejo de bases de datos *SQL*. Además es necesario conocer el funcionamiento de la herramienta *Eclipse* que permite trabajar con ese lenguaje y con el manejo del sistema de gestión de las bases de datos *MySQL*.

Una de las tareas fundamentales desarrolladas en este trabajo es la posibilidad de realización de grupos de trabajo de manera intuitiva para las distintas actividades que el profesor cree en un curso. Además de esto, los grupos de trabajo pueden ser calificados sin la necesidad de que un profesor haga uso del libro de calificaciones de manera individual para cada alumno. Esto supone una enorme mejora a la hora de trabajar con las distintas actividades que un curso puede poseer.

Gracias a este trabajo desarrollado se permite a los alumnos participar de manera activa en la formación de grupos de trabajo pudiendo de esta manera descargar al profesor de tarea y así fomentar a los estudiantes a colaborar en el curso de forma constante.

Por esto y mucho más, el módulo *Blended* es una buena herramienta de trabajo además poseer una gran flexibilidad que permite que el módulo se acerque a las necesidades tanto de profesores como de alumnos.

Capítulo 7. Líneas futuras

Tras todo el trabajo desarrollado, es posible plantear trabajo futuros que pueden ser realizados en el módulo:

- Añadir funcionalidad que permita introducir grupos de trabajo no solo con los códigos de identificación de los alumnos, si no con los nombres y/o apellidos de los mismos.
- A la hora de crear grupos de trabajo que sea posible introducir los miembros seleccionándolos de una lista sin tener que introducirlos de manera manual en los equipos.
- A la hora de calificar los equipos que el profesor pueda calificar a un alumno de manera individual dentro de un equipo si lo desea sin la necesidad de ir al libro de calificaciones.
- Permitir a un profesor que pueda imprimir varias páginas de tarea al mismo tiempo de distintos alumnos.
- Restringir el acceso a los alumnos a la hora de gestionar los equipos cuando son líderes de estos, ya que en un equipo que ya haya sido calificado por el profesor, el alumno no debería poder eliminar miembros de éste.
- Eliminación de tablas innecesarias a la hora de la realización del módulo que relacionan los grupos de *Moodle* con las tablas del módulo.

Bibliografía y referencias

- [1] Documentación Oficial de *Moodle*:
<http://docs.moodle.org/>, 2014
- [2] Documentación de la Wikipedia:
<http://es.wikipedia.org/wiki/Moodle>, 2014
- [3] Requisitos de *Moodle*:
http://docs.moodle.org/all/es/Instalaci%C3%B3n_de_Moodle#Requisitos, 2014
- [4] Requisitos de software de *Moodle*:
http://docs.moodle.org/all/es/Requisitos_para_instalaci%C3%B3n_de_moodle_2.6, 2014
- [5] Modulo básico de *Moodle*:
http://docs.moodle.org/dev/NEWMODULE_Documentation#Base_module_folder_content, 2014
- [6] Descripción de los ficheros de un módulo de *Moodle*
http://docs.moodle.org/dev/Activity_modules#Introduction, 2014
- [7] Estructura de una tabla de *Moodle* de la base de datos:
http://docs.moodle.org/all/es/Manual_de_Estilo_de_C%C3%B3digo#Estructuras_de_la_base_de_datos, 2008
- [8] Palabras reservadas de la base de datos:
http://docs.moodle.org/27/en/Database_reserved_words, 2009
- [9] Manual de *PHP*:
<https://php.net/manual/es/>, 2014

