



Universidad Autónoma del Carmen

Dependencia Área Ciencias de la Información

**Gestión de notificaciones de trabajos escolares
para un ambiente virtual de aprendizaje (AVA)
hacia dispositivos móviles con Android.**

T E S I S

Que para obtener el grado de
**Maestría en administración de tecnologías de la
información**

Presenta:

Marcelino Del Jesús Martínez Reyes

Director de tesis:

Dr. José Felipe cocón Juárez

Asesor:

M.C. Víctor Hugo Hernández Hernández

Cd. del Carmen, Campeche; Mayo de 2015.

DEDICATORIA

Para Dios y mis padres:

En primer lugar quiero agradecerle al creador de los cielos por haberme dado la vida y la oportunidad de poder ver un sueño más cumplido. Una prueba más de tu fidelidad, el haberte conocido ha sido lo mejor que me ha pasado, ya que si no hubiera sido por ti no sé dónde estaría ahora. No tengo como agradecerte lo mucho que me has dado, lo único que puedo decir es que te necesitaré siempre en mi vida.

A mi padre Marcelino Martínez Hernández por su apoyo desde mi infancia hasta el último día de su partida, porque jamás dejó de creer en mí y hoy lo que soy se lo debo a él, ya que hasta el último día supo ser un padre ejemplar y siempre será mi ejemplo a seguir.

A mi madre Violeta Reyes Arias por su apoyo y amor incondicional, gracias porque has estado en los momentos buenos y malos, jamás me has dejado y siempre has creído en mí, te amo.

A mis hermanos, a mi novia Addi Pacheco Ortiz, por cada palabra de aliento y de apoyo siempre has creído en mí, de igual manera a todos los que me han apoyado, Gracias.

AGRADECIMIENTOS

En primer lugar a Dios que me ha ayudado,
hasta el día de hoy, a mis padres, hermanos
y a mi novia.

De igual manera a mis asesores de tesis
por el apoyo dado en todo momento al
Dr. José Felipe Cocón Juárez y al profesor
M.C.C Víctor Hugo Hernández Hernández,
por su apoyo en la parte de *Android*.

RESUMEN

En este documento de tesis se aborda la implementación de una aplicación para dispositivos con sistema operativo *Android*, esta aplicación, consiste en que desde una plataforma educativa como es *moodle*, el profesor sube alguna tarea y en ese instante cuando carga la tarea, la plataforma realiza el envío de un mensaje o notificación al celular o dispositivo del alumno que se encuentra registrado en el curso, alertando al estudiante que existe una tarea para realizar, esto ayuda al alumno para permanecer informado sobre las actividades a realizar.

El documento consta de 5 capítulos en lo que se aborda la problemática a resolver, la propuesta de solución, el estado del arte, el modelo conceptual para el proyecto, incluyendo los tipos de tareas con la que cuenta *moodle* y en las cuales se hará el envío del mensaje, de igual manera la parte de la implementación es sobre el desarrollo de la aplicación en *Android*, y la edición para que *moodle* pudiera realizar el envío de la notificación al dispositivo.

Finalmente la parte de conclusiones y trabajos futuros, en donde se muestra lo que obtuvimos con la aplicación, además de las actividades que tenemos en mente y anexados para nuestra aplicación.

ABSTRACT

In this thesis document the development of an application for devices with Android operating system is addressed, this application is that from an educational platform like Moodle, the teacher uploads a task and in that instant when the task load platform performs he sent a message or notification to the phone or device, the student is registered in the course, alerting student that there is a task to perform, it helps the student to be not coming all the time to the Platform to see if there any pending.

The document consists of 5 chapters in the development problem, the proposed solution, the art, the conceptual model for the project, including the types of tasks that counts moodle and in which he will address sent the message, just as part of the implementation is on application development on Android, and the part that Annex to moodle could make the sending of the notification to the device.

Finally conclusions and part of future work, where shows what we got with the application, in addition to work in the future we have in mind to attach to your application.

Índice	Páginas
Dedicatoria	3
Agradecimientos	4
Resumen	5
Abstract	6
CAPÍTULO 1. INTRODUCCIÓN	11
1.1 Antecedentes	12
1.2 Planteamiento del problema.....	14
1.3 Justificación.....	15
1.4 Propuesta de solución.....	16
1.5 Objetivo General	17
1.5.1 Objetivos específicos.....	17
1.6 Alcances y Limitaciones del Proyecto	17
1.7 Definición de Recursos	18
1.8 Metodología de trabajo (diseño, procedimiento, análisis)	20
1.9 Plan de Actividades.....	21
1.10 Alcance.....	22
CAPÍTULO 2. ESTADO DEL ARTE	23
2.1 Introducción	24
2.2 Elementos de un sistema de <i>e-learning</i>	25
2.3 Evolución de las plataformas de <i>e-learning</i>	27
2.4 Los CMS, y su evolución hacia el <i>e-learning</i>	27
2.5 Características de las plataformas de <i>e-learning</i>	29
2.6 Herramientas de las plataformas de <i>e-learning</i>	31
2.7 Tipos de Plataformas educativas (comerciales)	32
2.8 Plataformas educativas (libres)	33
2.9 <i>Moodle</i> como plataforma educativa.....	35

2.9.1 Características de <i>moodle</i>	36
2.9.2 Materiales o Recursos con que cuenta <i>Moodle</i>	37
2.10 Uso de los Dispositivos móviles en la educación <i>on-line</i>	42
2.11 Aplicaciones <i>Moodle</i> para dispositivos móviles	43
2.12 <i>Android</i> como sistema operativo para plataformas educativas <i>on-line</i>	44
2.13 Características de <i>Android</i>	45
2.14 Ejemplos de aplicaciones de <i>Android</i> a <i>Moodle</i>	46
CAPÍTULO 3. MODELO CONCEPTUAL	48
3.1 Introducción	49
3.2 Las actividades de Moodle.....	49
3.3 Estructura del módulo de Actividades o evaluación	50
3.4 Módulo de tareas	51
3.4.1 Subir un solo archivo.....	53
3.4.2 Uploading avanzada de archivos	53
3.4.3 Texto en línea	53
3.4.4 Actividad desconectada.....	53
3.5 Modelo conceptual de la actividad (Tareas)	54
CAPÍTULO 4. IMPLEMENTACIÓN Y RESULTADOS	56
4.1 Introducción	57
4.2 Modelo de clases del módulo de tareas.....	57
4.3 Diagramas de Casos del módulo de (Tareas).....	59
4.4 Diagramas de robustez, para subir (Tareas).....	61
4.5 Diagramas de interacción, para el módulo de (Tareas).....	64
4.6 Implementación de las interfaces de <i>Android</i>	67
4.6.1 Creación del proyecto <i>Android</i> y registro en Google Cloud Mensajería (GCM).....	67
4.6.2 Desarrollo de las pantallas en <i>Android</i> (eclipse).....	72
4.6.3 Pantalla de conexión al servidor.....	73

4.7 Creación de la tabla en la base de datos (moodle)	74
4.8 Creación y ejecución del proyecto PHP.....	74
4.9 Implementación o modificación en moodle	79
4.9.1 Consulta de base datos (probándolo en la vista del curso)	80
4.10. Resultado de las pruebas	84
4.10.1 Interfaz del registro de usuario	84
4.10.2 Interfaz de conexión al servidor, en espera de mensajes	84
4.10.3 Interfaz del dispositivo, recibiendo una notificación	85
4.10.4 Interfaz del dispositivo, llegada de la notificación	86
4.10.5 Interfaz de moodle, donde se elige el tipo de tarea	86
4.10.6 Interfaz de moodle, donde se realiza la carga y el envío del mensaje	87
4.10.7 Interfaz de moodle, tarea subida.....	88
CAPÍTULO 5. CONCLUSIÓN. TRABAJO FUTURO	89
5.1 Conclusión.....	90
5.2 Trabajo a futuro.....	91
Referencias	92
Índice de figuras.....	94
Índice de tablas.....	96

CAPÍTULO 1

INTRODUCCIÓN

1.1 Antecedentes

Sin duda alguna la educación en México ha sido uno de los pilares más sobresalientes desde hace muchos años, y el hablar de ella es abarcar tanto la forma de enseñar como los métodos para hacerlo; mismos que al paso de los años estos han cambiado, la educación siempre ha sido de una forma presencial en la que el alumno tiene que ir a un aula educativa y el profesor es la persona encargada de proporcionar los conocimientos al estudiante, esta forma implica que hay un contacto físico entre el alumno y el profesor.

Hoy en día, existen algunas formas de enseñar a través de la educación a distancia, sin embargo, existen dos figuras en este tipo de educación, por una parte, el alumno se conecta desde algún lugar ya sea casa, trabajo, escuela u oficina, para desarrollar sus actividades escolares, por otra parte, el profesor es la persona que se encarga de proporcionar la orientación y de aclarar las dudas, sin embargo, el alumno investiga por su propia cuenta, pero no existe un contacto físico entre las dos partes.

La evolución de la educación a distancia se ha transformado en algo llamado *e-learning*, que combina lo mejor de la educación presencial y de la educación a distancia; ya que el profesor, desde una plataforma educativa vía *on-line* se conecta y hace uso de las herramientas que estas poseen.

En este sentido, las plataformas educativas o plataformas virtuales de aprendizaje constituyen un conjunto de estructuras, normas, técnicas, elementos y estrategias que se integran en la implementación del proceso enseñanza-aprendizaje (Sanchez Rodriguez, 2009), forma a su vez, parte esencial de los Ambientes Virtuales de Aprendizaje (AVA), que también son conocidos como EVA y VLE, esta última por su siglas del inglés de *Virtual Learning Environment*.

Un “Ambiente Virtual de Aprendizaje” o “Sistema Administrador del Aprendizaje” está diseñado para actuar como centro de las actividades de los estudiantes, para su

administración y facilitación, junto con la disposición de los recursos requeridos para ellas. Stiles nos explica que un ambiente virtual de aprendizaje (LMS por sus siglas en inglés) es equivalente a un sistema de administración del aprendizaje (Miranda Díaz, 2004).

Los ambientes virtuales de aprendizaje o plataformas educativas cuentan con algunas herramientas, las que ayudan para el desarrollo de actividades de aprendizaje, algunos de estos materiales, que la mayoría de estas plataformas contienen son los siguientes (Sanchez Rodriguez, 2009):

- Herramientas de distribución de contenido.
- Herramientas de comunicación, tales como el chat, foros, mensajería interna del curso, wikis, diarios.
- Herramientas de seguimiento y evaluación como son los cuestionarios, reportes, exámenes en línea, crucigramas, sopa de letras, actividades de relaciones, etc.
- Herramientas complementarias como portafolios, bloc de notas.

En la Internet existe un número considerable de estos sitios que ofrecen las gestiones de aprendizaje, que van desde software comerciales a software de código abierto, en este caso, los sistemas software más utilizados son los sistemas de gestión del aprendizaje (*Learning Management Systems*) o LMS. Como ejemplo de LMS de código abierto podemos mencionar *Moodle*, *LRN* o el reciente *Sakai* y, entre los sistemas comerciales, el más extendido es *Blackboard-WebCT*, *e-College* o *Desire2Learn* (Fernandez Pampillón, 2007).

Las plataformas tecnológicas utilizadas en la educación a distancia han evolucionado considerablemente, a tal grado que ahora ofrecen servicios de adaptabilidad y accesibilidad de contenidos en dispositivos móviles.

1.2 Planteamiento del problema

En la actualidad unos de los problemas que se presentan al seguir con los estudios académicos es la falta de recursos económicos, la falta de tiempo, entre otros, no obstante, en estos días han surgidos algunos métodos o formas de estudios, entre ellos está el asistir a clases de una forma *on-line*. La formación *on-line* consiste en estar inscrito en alguna escuela que proporcione estos cursos en la cual el profesor sube el material, las tareas o actividades a realizar, pero no existe un contacto físico con el docente, esto conlleva a que no se tenga que asistir a algún centro educativo, y de igual manera, el estudiante debe organizar sus tiempos, logrando trabajar y estudiar.

Esta modalidad de estudio es idónea para aquellas personas que trabajan, ya que no tendrían que asistir a un centro de estudio, solo se requiere estar conectado vía internet desde cualquier lugar se puede asistir a clases, no descuidando el trabajo. Actualmente, la Universidad Autónoma del Carmen cuenta con este modelo de educación a distancia impartiendo el programa educativo de licenciatura en tecnología de información (LTI) y el posgrado de Maestría en Administración de Tecnologías de Información (MATI), dicha institución cuenta con una plataforma educativa en la que imparte estos programas de forma *on-line*, y en la que el profesor hace uso de algunas herramientas educativas como foros, chat, videoconferencias, cuestionarios, entre otros, además el profesor sube a esta plataforma materiales de apoyos tales como libros, ligas web, que el alumno debe consultar o descargar estos materiales desde la plataforma para poder realizar las tareas y/o actividades puestas por el mentor. Todo esto se visualiza siempre y cuando se acceda a la plataforma educativa.

Consciente de la comunicación entre el sistema y el estudiante, el alumno a través de la plataforma educativa (AVA) realiza actividades marcadas por el profesor, y por un lado, el facilitador organiza los materiales de apoyo, lecturas, foros, actividades del curso en la plataforma educativa, y el alumno, revisa las actividades que fueron subidas o

montadas (*upload*) por el profesor, mismas que para realizarlas, el alumno tiene que entrar a la plataforma, es decir, el problema a resolver tiene que ver con el asunto de la plataforma que se maneja para la impartición de clases a nivel distancia (AVA). Cuando un profesor sube alguna tarea solo se sabe o el estudiante se da cuenta, cuando él ingresa a la plataforma para checar si hay algún pendiente, pero no existe un mecanismo que te avise en el instante que el profesor subió o actualizó alguna actividad, y viceversa, la problemática que se presenta es por la falta de comunicación entre el profesor y el estudiante, ya que el alumno no sabe cuándo ha surgido alguna actividad de dicha materia o asignatura en cuestión.

En estos momentos, la universidad no cuenta con este servicio ya que solo se sabe que existe una actividad siempre y cuando se acceda al portal, pero no existe algo que notifique en el instante quien actualiza los materiales o las actividades, ni mucho menos hacia un dispositivo móvil.

1.3 Justificación

La necesidad de proporcionar la comunicación en la plataforma educativa se da al facilitar las notificaciones de los profesores y alumnos, esta comunicación que permite fortalecer el acercamiento más estrecho entre estos actores importantes en el aprendizaje, logrando así un mejor control en las actividades marcadas por el profesor, ya que actualmente las plataformas educativas no facilitan este proceso, simplemente se limitan a que cada actor deba entrar a la plataforma para visualizar sus actividades o eventos marcados.

La realidad en que vivimos actualmente se encuentra alrededor de tecnologías inalámbricas de comunicación, llámese los dispositivos móviles, tales como tabletas (*tablets*), teléfonos móviles inteligentes (*Smartphone*), iPad, iPhone, entre otros. Estos dispositivos permiten una comunicación síncrona o asíncrona con las plataformas educativas, ya que cualquier dispositivo móvil puede acceder a la Internet, sin embargo, en

estos momentos no se cuenta con ninguna herramienta que permita una comunicación síncrona entre el profesor y estudiante mediante una actividad. La intención de facilitar esta comunicación se centra en el acercamiento del profesor-actividad-estudiante, con este enfoque la distancia marcada por la plataforma se transforma a una proximidad más palpable por los estudiantes, ya que estarán notificados de los eventos que programe el profesor mediante sus dispositivos móviles.

Otras de las potencialidades visibles, es el acercamiento entre profesor-estudiante, ya que cualquier estudiante cuenta con un medio de comunicación móvil, lo que facilita la comunicación entre profesor-estudiante para fines educativos, y más para facilitar el tiempo de organización de los estudiantes, para concluir satisfactoriamente sus actividades marcadas.

1.4 Propuesta de solución

La propuesta consiste en realizar una implementación adicional a moodle y una aplicación que gestione las notificaciones de las actividades propuestas por el profesor mediante el uso dispositivos móviles, es decir, el seguimiento de las actividades propuestas por el profesor serán enviadas a los estudiantes mediante el gestor de notificaciones (*Google Cloud Messagingfor Android –GCM*) implementado en una plataforma educativa, también conocida como AVA, EVA y VLE, por tanto, el profesor será el moderador de dicha actividad.

En concreto, se implantará una aplicación de notificaciones para que permita llevar el control de las actividades propuestas por el profesor y que genere las notificaciones de los eventos suscitados, por el docente, mediante dispositivos con *Android*. Además de efectuar modificaciones adicionales a moodle para poder hacer esta acción.

1.5 Objetivo General

Investigar y desarrollar un programa que facilite el envío de notificaciones a dispositivos móviles con sistema operativo *Android*, para las actividades desarrolladas por el profesor en la plataforma educativa *moodle*.

1.5.1 Objetivos específicos

- Investigación de plataformas educativas dentro del marco *opensource*.
- Exploración del mapa de actividad de la plataforma propuesta.
- Investigación y desarrollo de los componentes de acción para el envío a dispositivos móviles.
- Implementación de la aplicación como prototipo para la notificación a dispositivos móviles mediante las modificaciones adicionales para la plataforma educativa propuesta.

1.6 Alcances y Limitaciones del Proyecto

- El programa se desarrollará solo para celulares con sistemas operativos de *Android*.
- Sólo se podrá ver las notificación que se reciban (no modificarlas, ni reenviarlas).
- Las notificaciones solo serán enviadas a los alumnos que estén registrados en la materia o curso específico.
- Sólo se podrán recibir las notificaciones si se está conectado a internet.

1.7 Definición de Recursos

Para este apartado los recursos técnicos con lo que contaremos o haremos uso para el desarrollo de la tesis son los siguientes, van desde el hardware hasta el software que estaremos utilizando:

Hardware

- Una computadora portátil (Laptop)
 - Marca: HP
 - Procesador : AMD A8-4555M APU @1.60GHZ
 - Memoria RAM: 8Gb
 - Sistema operativo: Windows 8 64 bits
 - Disco Duro: 672Gb
- Un dispositivo móvil o Tablet
 - Marca: Samsung Galaxy NOTE 10.1
 - Sistema operativo: Android 4.0
 - Memoria: Micro SD hasta 32 Gb

Software

Estaremos utilizado una aplicación web de tipo Ambiente Educativo Virtual, de distribución libre, que ayuda a crear comunidades de aprendizaje en línea. Este tipo de plataformas tecnológicas también se conoce como LMS (Learning Management System). Y será el llamado *MOODLE* en su versión la 2.0 el cual está basado en lenguaje PHP.

Este software o plataforma es para agregar los módulos correspondientes, en el estarán los cursos en línea y sobre él se implementará lo de la notificación. Promoviendo el envío al dispositivo receptor.

Para el desarrollo de la aplicación del programa que estará en el dispositivo móvil, se estará utilizando el programa.

- Eclipse Classic
- En sus versión V3.3.1
- Un Instalador llamado Android SDK.

Base de datos

La base de datos que se estará utilizando para esta aplicación será la siguiente:

- MYSQL
- SERVIDOR APACHE

1.8 Metodología de trabajo (diseño, procedimiento, análisis)

En este proyecto la metodología de investigación se desarrolla de manera incremental mediante las siguientes etapas: Estado del Arte, Modelo Conceptual e Implementación.

1. Estado del arte

En esta primera etapa se desarrolló el tipo de investigación exploratoria, es decir, nos permite indagar sobre las nuevas aplicaciones y servicios que ofrecen las plataformas educativas, así como los avances tecnológicos en esta área. Se examinaron los antecedentes tanto de la plataforma educativa *moodle*, como la del sistema operativo *Android*, algunos de los temas que se analizaron para *moodle* fueron, los recursos con los que cuenta, como se realizaban cuentas de usuarios, creación de cursos y los permisos que se le daba a cada usuario.

En el caso de *Android*, los temas fueron, acerca del servicio de GMC (*Google Cloud Messaging for Android*), cómo funcionaba y qué se usa para poder realizar los envíos de las notificaciones a los dispositivos *Android*.

2. Modelo-conceptual

En esta segunda etapa, se desarrolló una investigación cualitativa, además, se realiza un análisis particular y el diseño conceptual de la propuesta de investigación, se procedió a construir los diferentes diagramas en donde se visualizan la estructura del módulo de *moodle* que se iba a modificar y los componentes que lo integran. De igual manera el diagrama de clases y sus campos.

3. Implementación

En esta etapa se realiza la implementación de manera individual, y además, se realizan pruebas experimentales con diversos dispositivos móviles con sistema operativo *Android*, tanto la interfaz de la aplicación, que es el que realiza el registro del dispositivo al servidor GMC, como también la parte de *moodle*, que se usa para realizar el envío de las notificaciones a los dispositivos que ya se registraron en el servidor, desde la aplicación.

1.9 Plan de Actividades

Este cronograma fue realizado con la ayuda del programa *Gantt Project*, está por mes empezando desde septiembre y terminando en octubre.

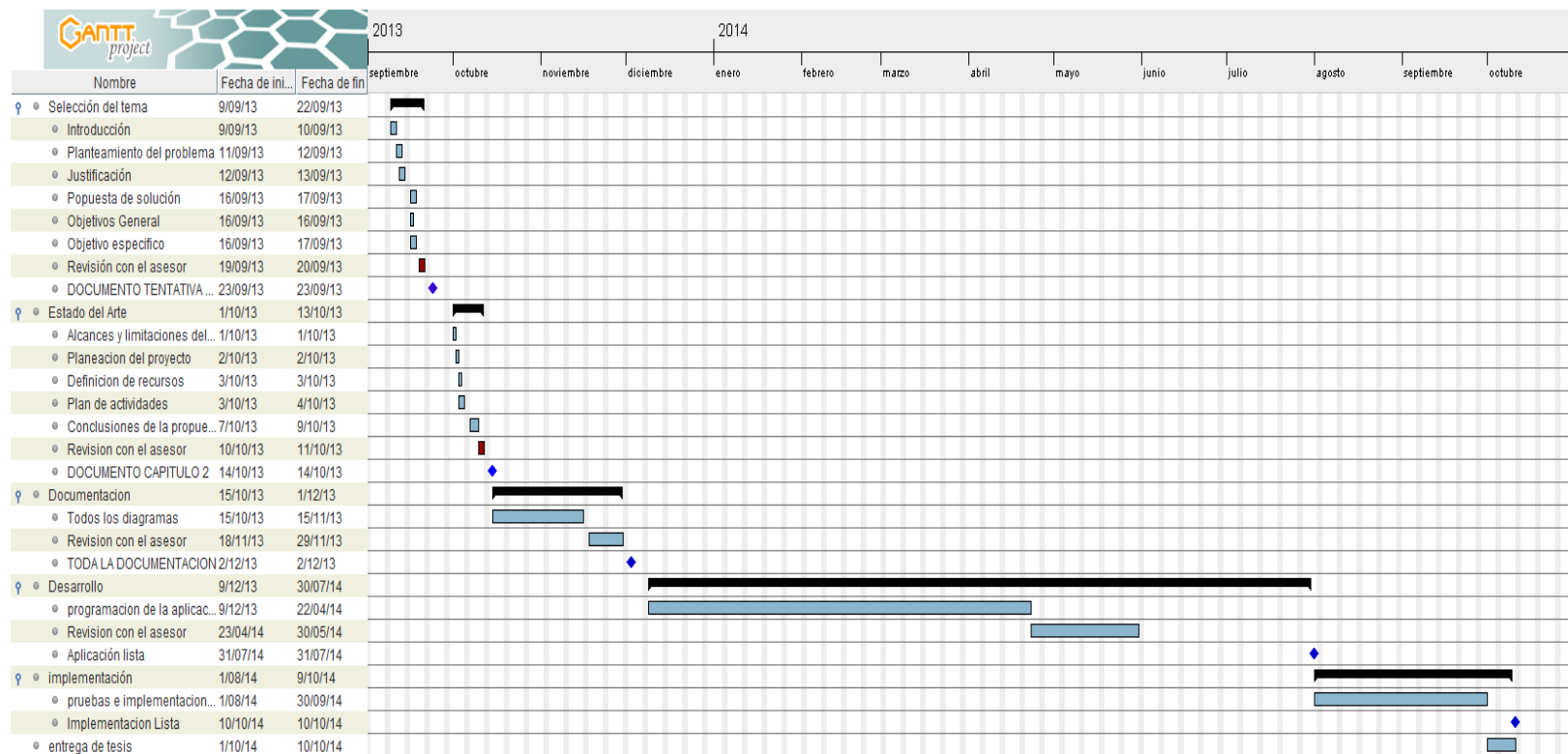


Figura 1. Cronograma de actividades.

1.10 Alcance

Como conclusión, se pretende realizar esta aplicación para beneficiar a los alumnos y a los profesores que están inscritos en alguna de las carreras de educación a distancia que ofrece la Universidad Autónoma del Carmen.

Ya que hoy en día los alumnos deben dar seguimiento a la hora y fecha, cuando el profesor sube alguna actividad o tarea, esto solo se sabe cuándo se entra a la plataforma educativa, pensando en esto optamos por realizar una aplicación que pueda mandar una notificación y/o aviso al celular a través de la plataforma *MOODLE*. Y con ello notificarle al alumno que el profesor acaba de subir una actividad, solo será el aviso.

Por otra parte, del lado del profesor, el subiría el material o las tareas y a la hora de subirla a la plataforma, la notificación llegara en tiempo real.

Para esta aplicación se tomarán los celulares que tienen sistema operativo en su versión ANDROID 4.0 y predecesores o antecesores.

CAPÍTULO 2
ESTADO DEL ARTE

2.1 Introducción

En los últimos tiempos, el desarrollo de la modalidad educativa a distancia tecnológica está poniendo en duda la propia diferenciación que ha habido tradicionalmente entre educación presencial y educación a distancia, e influye y hace replantearse muchos de los procesos educativos que se están dando actualmente en las aulas presenciales de educación superior, especialmente cuando se propone la combinación del uso del aula presencial y del aula virtual, lo que se ha convenido en denominar de forma generalizada aprendizaje combinado (*e-learning*).

En este escenario formativo combinado, surge un conjunto de cuestiones por resolver de gran relevancia para la calidad de los procesos educativos que tienen lugar en esta modalidad, como son el manejo tecnológico del aula virtual, las competencias tecnológicas del profesorado y el alumnado, la gestión del espacio y el tiempo educativo, el diseño de los contenidos y el tipo de actividades formativas (Barberá & Badia, 2005).

El *e-learning* puede ser clasificado de diferentes formas según los medios tecnológicos de los que hace uso, los cuales se corresponden con la evolución tecnológica que ha experimentado el aprendizaje a distancia basado en las TICS:

- El CBT (*Computer Based Training*) o CAI (*Computer Assisted Instruction*), aprendizaje basado en computador o instrucción asistida por computador, fue implantado en múltiples instituciones educativas y organizaciones. Estaba basado en la lectura e incorporaba mecanismos de realimentación pregunta-respuesta, convirtiendo al alumno en un ente más activo dentro de su propio proceso formativo.
- El IBT (*Internet Based Training*) fue el siguiente paso evolutivo de los sistemas de aprendizaje basados en computador, CBT. Con la llegada de Internet los contenidos podían llegar a sus destinatarios a través de Internet o de la intranet.

- El WBT (Web Based Training) consiste en el aprendizaje haciendo uso de la web, a través de la que se reciben los contenidos. En este último tipo se encuentra el campus virtual (Boneu, 2007).

Tabla 1. Comparativa entre sistemas CBT y sistemas e-learning.

Característica	Sistemas CBT	Sistemas de e-learning
Modelo pedagógico	Centrado en el profesor	Centrado en el alumno
Finalidad	Distribuir contenidos	Distribuir contenidos y capturar conocimiento
Tipo de componente educativo	Curso completo	Trozos de contenido u objetos de aprendizaje
Creación de contenido	Desde cero	Reutilización de contenido
Tiempo requerido para el aprendizaje	Días, semanas, meses	Horas

El *e-learning* es una forma de utilizar la tecnología para distribuir materiales educativos y otros servicios, permitiendo establecer un canal de retorno entre profesores y alumnos. En los nuevos entornos de aprendizaje se utiliza la tecnología web como la opción de distribución preferida en la actualidad, tanto para la distribución a través de una intranet como Internet.

2.2 Elementos de un sistema de *e-learning*

Los sistemas de comunicación pueden ser síncronos o asíncronos. Los sistemas síncronos son aquellos que generan comunicación entre usuarios en tiempo real, como podrían ser los chats o las videoconferencias. Los sistemas asíncronos no generan comunicación en tiempo real, pero ofrecen la posibilidad de que las aportaciones de los usuarios queden grabadas. El correo electrónico y los foros son algunas de las herramientas que usan este tipo de comunicación (Boneu, 2007):

- Las plataformas de *e-learning* son el software de servidor que se ocupa principalmente de la gestión de usuarios, cursos y de la gestión de servicios de comunicación.
- Los contenidos o *courseware* es el material de aprendizaje que se pone a disposición del estudiante. Los contenidos pueden estar en varios formatos, en función de su adecuación a la materia tratada. El más habitual es el WBT, cursos en línea con elementos multimedia e interactivos que permiten que el usuario avance por el contenido evaluando lo que aprende.

Para la reutilización e interoperabilidad de contenidos en diferentes plataformas, debe cumplirse una doble premisa: por un lado los cursos deben seguir un estándar y por otro lado las plataformas deben soportar dicho estándar, con lo que se facilita el uso de cursos realizados por la propia organización y por terceros.

Hoy en día no existe un único estándar en el mercado, pero todos intentan solucionar estos problemas de forma independiente. El estándar LOM de IEEE LTSC ampliamente aceptado, permite describir el contenido de un objeto de aprendizaje a través de metadatos. La AICC fue el primer organismo creado para desarrollar un conjunto de normas que permitieran el intercambio de cursos CBT. La descripción de itinerarios formativos es posible mediante las estructuras de descripción de recursos RDF o bien a través de IMS LD, y el estándar SCORM de ADL, ampliamente aceptado y utilizado, permite la organización de contenidos soportando la descripción de itinerarios formativos, secuenciación de contenidos, el empaquetamiento de los contenidos para su cómoda distribución, y el seguimiento del proceso de aprendizaje. La integración de los diferentes estándares en *SCORM* se presenta en el siguiente gráfico (Boneu, 2007).

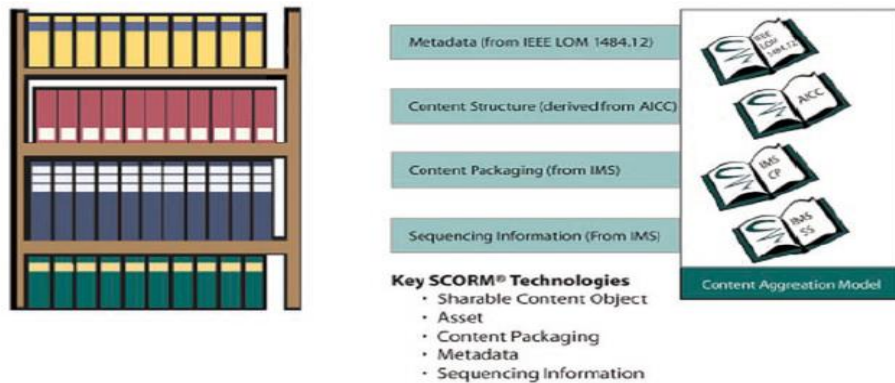


Figura 2. El libro modelo de agregación de contenidos SCORM como parte de la librería

2.3 Evolución de las plataformas de *e-learning*

El origen de las plataformas de *e-learning*, basadas en web, se debe a una especialización de los CMS, sistemas de gestión de contenidos, en sistemas orientados a la gestión de contenidos para el aprendizaje a distancia.

2.4 Los CMS, y su evolución hacia el *e-learning*

El sistema de gestión de contenidos (*Content Management Systems* o *CMS*) es un software que se utiliza principalmente para facilitar la gestión de webs, ya sea en Internet o en una intranet, y por esto también son conocidos como gestores de contenido web (*Web Content Management* o *WCM*). Cabe tener en cuenta, sin embargo, que la aplicación de los CMS no se limita sólo a las webs, y en el caso del *e-learning* la gestión no está centrada en la web, sino en los contenidos educativos (RLOs, recursos, documentos y pruebas evaluadoras, entre otros). La evolución de los CMS hacia los sistemas *e-learning* tiene unas necesidades concretas que un CMS general no siempre cubre, o si lo hace, no da las mismas facilidades que una herramienta creada para realizar esta función (Boneu, 2007).

- Primera etapa: los CMS (*Content Management System* o *Course Management System*) son dentro de las plataformas de *e-learning* los más básicos y permiten la generación de sitios web dinámicos. El objetivo de estos programas es la creación y gestión de información en línea (textos, imágenes, gráficos, vídeos, sonido, etc.). También se caracterizan por no poseer herramientas elaboradas de colaboración (foros, chats, diarios, etc.) ni apoyo en tiempo real.
- Segunda etapa: los LMS (*Learning Management System*) aparecen a partir de los CMS y proporcionan un entorno que posibilita la actualización, mantenimiento y ampliación de la web con la colaboración de múltiples usuarios. Están orientados al aprendizaje y la educación, proporcionando herramientas para la gestión de contenidos académicos, permitiendo mejorar las competencias de los usuarios de los cursos y su intercomunicación, en un entorno donde es posible adaptar la formación a los requisitos de la empresa y al propio desarrollo profesional. Disponen de herramientas que permiten la distribución de cursos, recursos, noticias y contenidos relacionados con la formación general.
- Tercera etapa: los LCMS (*Learning Content Management System*) son plataformas que integran las funcionalidades de los CMS y los LMS, que incorporan la gestión de contenidos para personalizar los recursos de cada estudiante y donde las empresas se convierten en su propia entidad editora, con autosuficiencia en la publicación del contenido de una forma sencilla, rápida y eficiente, resolviendo los inconvenientes de las anteriores plataformas. Ofrecen facilidad en la generación de los materiales, flexibilidad, adaptabilidad a los cambios, control del aprendizaje y un mantenimiento actualizado del conocimiento (Boneu, 2007).

Tabla 2. Resumen comparativo de los LMS y LCMS

Usos	LMS	LCMS
Usuarios a los que va dirigido	Responsables de los cursos, administradores de formación, profesores o instructores	Diseñadores de contenidos, diseñadores instruccionales, directores de proyectos
Proporciona	Cursos, eventos de capacitación y está dirigido a estudiantes	Contenidos para el aprendizaje, soporte en el cumplimiento y usuarios
Manejo de clases, formación centrada en el profesor	Sí (pero no siempre)	No
Administración	Cursos, eventos de capacitación y estudiantes	Contenidos para el aprendizaje, soporte en el cumplimiento y usuarios
Análisis de competencias-habilidades	Sí	Sí (en algunos casos)
Informe del rendimiento de los participantes en el seguimiento de la formación	Enfoque principal	Enfoque secundario
Colaboración entre usuarios	Sí	Sí
Mantiene una base de datos de los usuarios y sus perfiles	No siempre	No siempre
Agenda de eventos	Sí	No
Herramientas para la creación de contenidos	No	Sí
Organización de contenidos reutilizable	No siempre	Sí
Herramientas para la evaluación integrada para hacer exámenes	Sí (la mayoría de los LMS tienen esta capacidad)	Sí (la gran mayoría tienen esta capacidad)
Herramienta de flujo de trabajo	No	Sí (en algunas ocasiones)
Comparte datos del estudiante con un sistema ERP (<i>enterprise requirement planning</i>)	Sí	No
Evaluación dinámica y aprendizaje adaptativo	No	Sí
Distribución de contenido, control de navegación e interfaz del estudiante	No	Sí

2.5 Características de las plataformas de *e-learning*

Hay cuatro características básicas, e imprescindibles, que cualquier plataforma de *e-learning* debería tener: (Boneu, 2007)

- **Interactividad:** conseguir que la persona que está usando la plataforma tenga conciencia de que es el protagonista de su formación.
- **Flexibilidad:** conjunto de funcionalidades que permiten que el sistema de *e-learning* tenga una adaptación fácil en la organización donde se quiere implantar. Esta adaptación se puede dividir en los siguientes puntos.

- Capacidad de adaptación a la estructura de la institución.
- Capacidad de adaptación, donde se quiere implantar el sistema.
- Capacidad de adaptación a los contenidos y estilos pedagógicos.

- **Escalabilidad:** capacidad de la plataforma de *e-learning* de funcionar igualmente con un número pequeño o grande de usuarios.
- **Estandarización:** hablar de plataformas estándares es hablar de la capacidad de utilizar cursos realizados por terceros; de esta forma, los cursos están disponibles para la organización que los ha creado y para otras que cumplen con el estándar. También se garantiza la durabilidad de los cursos evitando que éstos queden obsoletos y por último se puede realizar el seguimiento del comportamiento de los estudiantes dentro del curso.

Otras características generales observables en las plataformas de *e-learning* son las siguientes, (Boneu, 2007).

- **Código abierto:** se habla de software (*Open Source*), cuando éste se distribuye con licencia para poder ver y modificar el código fuente base de la aplicación.
- **Plataforma gratuita:** el uso de la plataforma no supondrá ningún coste por adquisición o licencia de uso. También existe el caso de las plataformas GPL (*General Public License*) Open Source, donde los desarrolladores de estas plataformas ofrecen apoyo en la instalación y otros servicios de manera comercial.
- **Internacionalización o arquitectura multi idioma:** la plataforma debería estar traducida, o se debe poder traducir fácilmente, para que los usuarios se familiaricen con ella.
- **Tecnología empleada:** en cuanto a la programación, destacan en este orden PHP, Java, Perl y Python, como lenguajes Open Source, muy indicados para el desarrollo de webs dinámicas y utilizados de manera masiva en las plataformas GPL.
- **Amplia comunidad de usuarios y documentación:** la plataforma debe contar con el apoyo de comunidades dinámicas de usuarios, con foros de usuarios, desarrolladores, técnicos y expertos.

2.6 Herramientas de las plataformas de *e-learning*

Las actuales plataformas de e-learning ofrecen muchas funcionalidades, que pueden ser agrupadas de la siguiente manera (Sanchez Rodriguez, 2009):

- Herramientas de distribución de contenidos. Para el profesorado debe proveer un espacio para colocar a disposición del alumnado información en forma de archivos (un repositorio de contenidos), que pueden tener distintos formatos (HTML, PDF, TXT, ODT, PNG) y que se pueden organizar de forma jerarquizada (a través de carpetas/ directorios). Debe disponer de diversas formas de presentar contenidos e información: enlaces a archivos, a páginas Web, calendarios, etiquetas con diversos elementos (texto, imágenes estáticas y en movimiento), FAQs, glosarios.
- Herramientas de comunicación y colaboración síncronas y asíncronas para que los participantes de una actividad formativa puedan comunicarse y trabajar en común: foros de debate e intercambio de información, salas de Chat, mensajería interna del curso con posibilidad de enviar mensajes individuales y/o grupales, wikis, diarios, formación de grupos de trabajo dentro del grupo-clase.
- Herramientas de seguimiento y evaluación, como cuestionarios editables por el facilitador para evaluación del alumno, autoevaluación para los mismos, tareas, reportes de la actividad de cada alumno(a), plantillas de calificaciones.
- Herramientas de administración y asignación de permisos, que posibiliten asignar perfiles dentro de cada curso, controlar la inscripción y el acceso (esto generalmente se hace mediante autenticación con nombre de usuario y contraseña para usuarios registrados), etc. Estos procedimientos se pueden hacer a nivel de administrador, pero también a nivel de profesorado (por ejemplo activar o desactivar un curso, inscribir a usuarios del sistema como profesorado o alumnado del mismo, cierta personalización del entorno).

- Herramientas complementarias. No estaría de más que la plataforma dispusiera de un portafolio, bloc de notas, sistemas de búsquedas de contenidos del curso y/o foros.

2.7 Tipos de Plataformas educativas (comerciales)

En la actualidad existe un número bastante amplio de plataformas, entre ellas encontramos las comerciales las cuales nos indican que son herramientas por las que hay que pagar una cuota de instalación y/o mantenimiento que suele variar en función del número de usuarios y que hay que renovar cada cierto tiempo (Sanchez Rodriguez, 2009).

Las plataformas comerciales han evolucionado rápidamente en su complejidad y han generado sucesivas versiones que incorporan herramientas y aplicaciones cada vez más versátiles, completas y complejas que permiten una mayor facilidad en el seguimiento de un curso virtual y en la consecución de los objetivos que pretende, tanto académicos como administrativos y de comunicación, (Garcia Sanchez & Castillo Rosas, 2005).

García Sánchez, Afirma que, en principio, pueden parecer la mejor opción para poner en funcionamiento acciones formativas de enseñanza a distancia en nuestra Universidad por varias razones:

- Suelen ser fáciles de instalar y estar bien documentadas el servicio de asistencia técnica suele ser ágil y rápido.
- Suelen estar muy testeadas por departamentos de control de calidad que llevan a cabo muchas pruebas. Ofrecen derecho a actualizaciones competitivas o a la adquisición del producto de por vida (con una cuota de mantenimiento anual).
- Dan alta fiabilidad. Suelen ofrecer una estabilidad contrastada. Hay empresas que diseñan y desarrollan módulos específicos que mejoran, apoyan o extienden los servicios prestados por las que tienen más penetración en el mercado.

No obstante, informa de sus inconvenientes y restricciones: a medida que se han ido asentando han aumentado el precio de las licencias. En muchos casos, sólo existen dos modelos de licencia:

- Completa, en la que el costo va en función del número total de alumnos y alumnas de la institución (y no en función del alumnado virtual de la misma).
- Limitada al número de alumnos y alumnas permitido. Una licencia generalmente da derecho a instalar la aplicación únicamente en un servidor (va asociada a una IP), lo que supone: Problema con las actualizaciones, ya que no pueden tener funcionando a la vez en equipos distintos una versión antigua y otra nueva de una misma aplicación.

Hay varias de ellas muy conocidas y extendidas como:

- Blackboard: <http://www.blackboard.com/>
- WebCT: <http://www.webct.com/> (adquirida por Blackboard en 2006)
- Virtual Profe: <http://www.ingenia.es/>
- e-training: <http://www.encyclomedia.es/>
- jenzabar: <http://www.jenzabar.net/>

2.8 Plataformas educativas (libres)

Al hablar de una plataforma de software libre es la que dispone de un tipo especial de licencia, denominada GPL (General Public License), que posibilita cuatro libertades a los usuarios (Sanchez Rodriguez, 2009):

- La libertad de usar el programa con cualquier propósito.
- la libertad de estudiar cómo funciona el programa y adaptarlo a tus necesidades.
- La libertad de estudiar cómo funciona el programa y adaptarlo a tus necesidades.

- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie.

El acceso al código fuente es una condición previa para esto. La libertad de distribuir copias, con lo que puedes ayudar a tu vecino, la libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie, el acceso al código fuente es un requisito previo para esto. Específica que aunque es software libre no es sinónimo de gratuidad, sino de libertad.

Este tipo de software para plataformas comparte las mismas ventajas (obviamente los mismos inconvenientes) que cualquier otro tipo de aplicaciones con este tipo de licencia: La posibilidad de acceder al código fuente hace que estas aplicaciones sean más “seguras”.

- Reducción o en su caso eliminación total, de costes. En la mayoría de las ocasiones no hay que pagar por actualizaciones ni por número de licencias.
- Posibilidad de reutilización de código entre aplicaciones.
- La decisión de evolución de funcionalidades de toda la comunidad de usuarios y no una empresa.
- La comunidad de usuarios supone un amplio “banco de pruebas”, con lo que cuando se liberan versiones suelen ser bastante estables.

Entre las plataformas más comunes de este tipo encontramos las siguientes:

- Bazaar: <http://klaatupc.athabascau.ca/>
- Claroline: <http://www.claroline.net>
- Moodle: <http://moodle.org/>
- ILIAS: <http://www.ilias.de/ios/>
- Dokeos: <http://www.dokeos.com/>
- Sakai: <http://www.sakaiproject.org/>

- Manhattan Virtual Classroom: <http://manhattan.sourceforge.net/>

El software libre suele ser muy modular, con lo que permite la posibilidad de instalar y ejecutar aquello que se necesita. Se podría indicar, como inconveniente, que las plataformas comerciales ofrecen más funciones. No obstante, cada vez más se van acercando unas a otras y las diferencias se van minimizando.

2.9 Moodle como plataforma educativa

Moodle ofrece como entorno virtual de enseñanza-aprendizaje, considerándolo en comparación con los anteriormente expuestos, mucho más viabilidad y eficacia para llevar a cabo procesos de formación *on-line*, (Llorente Cejudo, 2007).

El caso de *Moodle* como EVE/A es destacable por varios motivos, tales como: por la comunidad de usuarios y desarrolladores que se ha creado en torno al mismo contribuyendo con constantes mejoras e innovaciones hacia un proceso de construcción colaborativo. Por ser una plataforma de código libre, por las posibilidades educativas que ofrece y los principios de su diseño y desarrollo continuo.

Moodle: hay unos 24500 sitios Web que la utilizan en 175 países. Está traducida a más de 75 idiomas. Teniendo en cuenta que Moodle fue iniciada en 1999 y que la primera versión salió en 2002, las cifras anteriores nos dan idea del crecimiento exponencial de esta plataforma. Va camino de convertirse en un estándar de plataforma educativa virtual, con usuarios tan prestigiosos como la británica *Open University*, (Sanchez Rodriguez, 2009).

Moodle fue creado en 1999 por Martin Dougia, computista y educador australiano. ¿Cómo se desarrolla esta filosofía educativa en las aulas virtuales creadas en Moodle? Para lograrlo, Moodle facilita el agregar recursos didácticos y actividades interactivas en sus aulas virtuales (Davila, 2011).

2.9.1 Características de *moodle*

Algunas de las características de esta plataforma de enseñanza son que funciona sobre Linux, Mac y Windows, todos los archivos están cifrados y se realizan continuas copias de seguridad automáticas de los cursos que impiden la pérdida de cursos, documentos y archivos. Los profesores pueden añadir una clave de acceso a los cursos lo que nos permite diferentes opciones como abrir el curso sólo a nuestros estudiantes, o compartir a invitados e incluso a otros profesores para trabajar y cooperar en nuestra asignatura (Martinez De Lahidalga, 2008).

Resulta fácil migrar de otras plataformas de aprendizaje o aplicaciones informáticas (Word, Power Point, Pdf) que estemos utilizando en la actualidad. Moodle dispone de una excelente documentación de apoyo en línea y comunidades de usuarios que pueden solucionar cualquier duda, por medio de los diferentes foros destinados a ello.

Tres son los grandes recursos de moodle:

- Gestión de Contenidos.
- Comunicación.
- Evaluación.

Para **gestionar los contenidos** empleamos el presentar al alumnado los apuntes de nuestro curso que podemos complementar con otros materiales como imágenes, gráficas o videos y también tendremos la oportunidad de entrar en otras páginas web relacionadas con el tema. Tiene un editor HTML “WYSIWYG” incluido, lo que nos permite a los usuarios, como alumnos o como profesores, además de escribir texto como tradicionalmente hacíamos en nuestros apuntes o trabajos, incluir o enlazar (link) las más variadas fuentes y recursos 2.0, como múltiples blogs, web-quest, imágenes, videos o documentos, que harán mucho más completo y variado el contenido.

Para **comunicarnos** con nuestros alumnos, moodle dispone de varias opciones siendo la más utilizada la de los foros, por medio de los cuales podemos gestionar las tutorías de manera individual o grupal, aspecto fundamental en la implantación de los ECTS.

La plataforma *moodle* facilita el aprendizaje cooperativo a través de estos foros en los que los propios alumnos dan respuesta a las preguntas y dudas generales planteadas por otros alumnos de su grupo, además de chat.

Por último la **evaluación** de los alumnos para la que se dispone de múltiples opciones en función de nuestro grado de implantación de las pedagogías más activas, de este modo podemos enviar tareas que estén en relación a las capacidades o competencias que tengan que acreditar los alumnos. También es factible preparar cuestionarios específicos por temas autoevaluables y con *feed-back* inmediato al alumno de sus resultados, incluso podemos hacer que los alumnos colaboren o se evalúen entre ellos usando el concepto y la herramienta de wiki, especial importancia adquiere moodle para los profesores de las ramas técnicas y científicas, contando con multitud de herramientas específicamente creadas para estas áreas.

2.9.2 Materiales o Recursos con que cuenta Moodle

Son documentos que pueden emplearse para hacer llegar al estudiante un conjunto de informaciones, orientaciones, la explicación de determinados contenidos, así como textos, artículos, entre otras lecturas. Se pueden editar directamente en Moodle o adjuntar un archivo de contenido digital, Word, Power Point, Flash, vídeo, sonidos, etc. Los archivos pueden subirse y manejarse en el servidor, o pueden ser creados sobre la marcha. Pueden enlazarse aplicaciones web para transferir datos (López Ramos).

Los diferentes tipos de materiales o recursos, básicamente especifican cómo es el contenido al que se va a acceder:

- * Texto Plano: el tipo más sencillo. Usted simplemente puede teclear (editar) la página en un formulario que aparece en ella. Algunas de las cosas que teclee serán automáticamente formateadas, de la misma forma que los mensajes en los foros. Vea la ayuda sobre escribir texto.
- * Texto HTML: para los usuarios más avanzados. No se efectúa ningún tipo de formateo automático, sino que usted tiene que hacerlo manualmente usando los códigos de HTML.
- * Referencia: una referencia a un periódico o libro.
- * Archivo subido: muestra cualquier archivo que haya subido al curso. Vea la sección de "Archivos".
- * Enlace Web: una URL (Dirección de Internet) a alguna parte de la web. Cuando alguien hace clic en este recurso será llevado fuera del curso y la página especificada llenará la ventana del navegador.
- * Página Web: una URL a alguna parte de la web. Como el tipo de recurso anterior, esto mostrará una página web. Pero a diferencia del anterior, la página se mostrará dentro de un marco, como si estuviera integrada dentro del curso.

Foros:

Es un intercambio asincrónico, es decir en tiempo no simultáneo, entre los que participan en el proceso educativo. A diferencia de las actividades que se realizan de manera sincrónica, quienes participan en un foro no están sujetos a un día y horario específico para comunicarse, sino que pueden acceder a la información puesta en el foro e incorporar otra según dispongan de tiempo para ello en los plazos establecidos para dicha actividad, lo que ofrece más flexibilidad y la posibilidad de hacer lecturas más reflexivas. El profesor puede orientar la suscripción de todos a un foro o dar la posibilidad de que cada persona elija de manera voluntaria a qué foros suscribirse, lo que depende de la finalidad de la tarea a realizar, además, también puede mover fácilmente los temas de discusión entre distintos foros.

Chat:

El módulo de Chat permite que los participantes discutan en tiempo simultáneo a través de Internet. A diferencia del foro la comunicación si está prevista para un día y hora específica y requiere de lecturas y respuestas más rápidas que no ofrecen muchas oportunidades para la reflexión. Usar una sala de Chat es bastante distinto a utilizar los espacios de un foro. Este es muy útil para tener una comprensión de los otros y del tema en debate, en particular cuando se trata de actividades no académicas y menos formales. Tanto en la sección de Chat como en otras actividades colectivas (Foros, talleres, etc.) se pueden considerar, sin violar estas normas elementales de respeto, la contribución al establecimiento de relaciones de amistad y de compañerismo entre los participantes lo que consolida la formación de un verdadero colectivo.

Taller:

El Taller es una actividad para el trabajo en grupo con un vasto número de opciones, en la que se puede insertar imágenes, tablas, enlaces, caritas expresivas, pegar textos de otras aplicaciones de Windows, etc. Permite a los participantes diversas formas de evaluar los proyectos de los demás, así como proyectos-prototipo. También coordina la recopilación y distribución de esas evaluaciones de varias formas.

La estrategia de calificación en una tarea del Taller es bastante flexible. Puede ser: ninguna calificación, calificación acumulativa y calificación referida a criterio entre otras. En este espacio los participantes establecen la discusión sobre uno o varios temas del contenido del curso, determinado previamente y dirigido por el profesor.

Diario:

Es un espacio en el que el estudiante puede colocar sus reflexiones y apuntes personales, lo que constituye una información privada, visible solo por él y su profesor tutor o coordinador. Le permite al estudiante guardar sus experiencias, notas y comentarios sobre los temas tratados, actividades desarrolladas o cualquier acontecimiento referido.

El profesor o tutor debe motivar la entrada en el diario con una pregunta abierta o sugerencia y puede evaluar al grupo completo con un único formulario por cada entrada particular al diario. Los comentarios del profesor se adjuntan a la página de entrada del diario y la notificación se envía por correo.

Glosario:

Esta actividad permite a los participantes crear y mantener una lista de definiciones, como un diccionario. Las entradas (incorporaciones) pueden buscarse (navegar) de diferentes maneras. Permite al profesor exportar las entradas de un glosario (el principal) a otro secundario dentro del mismo curso. Para hacer esto, usted debe especificar cuál glosario es el principal. Puede tener sólo un glosario principal por curso, que sólo los profesores pueden actualizarlo. Las entradas exportadas a un glosario principal sólo pueden actualizarse o anularse por profesores y en cualquier momento. En el caso de los glosarios secundarios los estudiantes pueden hacerlo, pero se puede elegir si esas entradas se incorporarán automáticamente o deben ser previamente aprobadas. Es posible permitir que otros agreguen comentarios a las entradas del glosario escogiendo habilitar o no esta posibilidad.

Lección:

Una lección entrega el contenido de una forma flexible e interesante. Consiste en un grupo de páginas, cada una de las cuales termina con una pregunta y un número de respuestas posibles. Dependiendo de la respuesta escogida por el alumno, continúa a la página siguiente o regresa a la página anterior. La navegación dentro de una lección puede ser simple o compleja, lo que está condicionado en gran medida por la estructura del material que se está presentado.

Cuestionarios:

Se conforman a partir de una base de datos de preguntas elaboradas por los profesores con anterioridad, que podrán ser reutilizadas para estructurar cuestionarios en diferentes ocasiones. Las preguntas pueden ser almacenadas en categorías de fácil acceso que pueden ser publicadas para hacerlas accesibles desde cualquier curso del sitio. Se califican automáticamente y es posible recalificarlas si se reenvían para su modificación.

El profesor puede precisar un límite de tiempo a partir del cual el cuestionario ya no estará disponible. De igual modo puede determinar si pueden ser resueltos varias veces y si se mostrarán o no las respuestas correctas y los comentarios a los estudiantes.

Tipos de cuestionarios:

- Opción Múltiple
- Respuesta Corta
- Numérica
- Verdadero/Falso
- Emparejamiento
- Preguntas incrustadas
- Emparejamiento aleatorio de respuesta corta
- Aleatoria

Encuestas:

Se trata de encuestas ya preparadas y contrastadas como instrumentos para el análisis y valoración de las actividades del proceso de enseñanza y aprendizaje en línea. Los profesores pueden usarlas tanto para recopilar datos de sus alumnos como de su propia enseñanza.

Las encuestas incluidas en Moodle se han elegido por ser particularmente útiles para la evaluación de entornos de aprendizaje online que siguen una pedagogía participativa, así como, para identificar ciertas tendencias que pueden estar dándose entre sus participantes. La interfaz de las encuestas impide la posibilidad de que estas sean respondidas sólo parcialmente.

2.10 Uso de los Dispositivos móviles en la educación *on-line*

Desde hace algunos años los avances tecnológicos han conseguido que el acceso a servicios y herramientas online sea cada vez más fácil y asequible en cualquier momento y en cualquier lugar. Los nuevos dispositivos móviles de acceso a las telecomunicaciones ponen al alcance de la palma de la mano lo que antes solamente estaba disponible desde un ordenador fijo con las consecuentes restricciones de movilidad. La capacidad de estar interconectado con tecnologías sin cables (*Wireless*) así como el desarrollo de dispositivos portátiles (*smartphone, tablets*) cada vez más versátiles y funcionales, posibilita unas facilidades de movilidad que han revolucionado la manera en la que los usuarios se relacionan con el mundo, (Arjona Heredia & Gámiz Sanchez, 2013).

Para realizar una aproximación al concepto y a las implicaciones del *Mobile learning* o *m-learning* se debe tener en cuenta que se trata de un concepto emergente que va evolucionando día a día y que, por ello, tenemos que ser conscientes de los procesos de cambio que pueden ir surgiendo en el futuro. Además los intentos que se han producido para su conceptualización parten de la base de reconocer que el *Mobile learning* es esencialmente personal, dependiente del contexto y situación, lo que lo hace especialmente problemático para su definición y evaluación. De una manera general, se puede decir que el *Mobile Learning* o *m-learning* es una forma de *e-learning* que emplea específicamente dispositivos de comunicación sin cables (*wireless*) para la distribución de contenidos y el apoyo del aprendizaje y para la adquisición de una interacción enriquecida del aprendizaje y la enseñanza en el entorno apropiado.

Si se entiende el *m-learning* como una evolución desde el *e-learning* que desarrolla procesos de aprendizaje a través de internet, se debe tener en cuenta algunas propiedades que diferencian un dispositivo móvil de un ordenador de escritorio y que tienen su impacto en lo que resulta razonable, útil e incluso cómodo hacer en este tipo de dispositivos. Algunas de esas características pueden ser la salida (por ejemplo el tamaño y resolución de pantalla); la entrada (por ejemplo el teclado o la pantalla táctil); la capacidad de procesamiento y memoria; o las aplicaciones y componentes soportados. De este modo, se deberán tener en cuenta esas limitaciones a la hora de transferir servicios proporcionados por una plataforma de *e-learning* para ser desarrollados en un entorno móvil, al igual que es necesario ser conscientes de que se ponen a nuestra disposición nuevos servicios dependientes de la movilidad que podremos incluir como son, por ejemplo, el nuevo contexto y la capacidad de localización.

Actualmente una de las plataformas de *e-learning* más extendidas a nivel mundial es el entorno de enseñanza virtual *Moodle*. La plataforma *Moodle* es una herramienta libre y gratuita basada en su diseño en el constructivismo pedagógico y que cuenta con una gran comunidad de seguidores que mantienen el proyecto activo.

2.11 Aplicaciones *Moodle* para dispositivos móviles

El soporte de *Moodle* para el *m-learning* ha ido mejorando versión tras versión, posibilitando que la comunidad realice desarrollos que permitan al usuario seguir conectado a la plataforma usando dispositivos móviles. Los primeros desarrollos fueron aplicaciones móviles nativas, estas aplicaciones deben ser programadas independientemente para cada sistema operativo móvil (*Symbian, iOS, Android, Blackberr*) y frecuentemente necesitan modificaciones o configuraciones específicas en el servidor adecuadas para su correcto funcionamiento, (Arjona Heredia & Gámiz Sanchez, 2013).

Sólo son capaces de mostrar algunas herramientas estándar de *Moodle* como por ejemplo foros, actividades simples, materiales y calendarios, por lo que no serían una alternativa válida cuando en un curso se hace uso intensivo de extensiones no incluidas en una instalación típica de *Moodle*, muchas de ellas no son autónomas, es decir, necesitan de un navegador para mostrar los contenidos de *Moodle* y otras basan su funcionamiento en la transformación de los contenidos web que reciben del servidor para adaptarlos a las características de los dispositivos móviles (*screen scraping*).

Los dispositivos móviles cuentan con una serie de funcionalidades cuyo aprovechamiento se ha extendido de forma exponencial en los últimos años y que pueden ser aplicadas en el ámbito de la enseñanza como son, por ejemplo:

- El uso de los sistemas de posicionamiento/ubicación (GPS, aGPS, brújula) que podrían dotar a la plataforma de sensibilidad al contexto del usuario.
- Uso de nuevos modelos de interacción con el dispositivo usando interfaces táctiles o cámaras integradas. No obstante, es muy poco frecuente encontrar aplicaciones en este terreno que hagan uso de las características específicas de los dispositivos móviles.

2.12 *Android* como sistema operativo para plataformas educativas *on-line*.

Un Sistema Operativo para dispositivos móviles es considerado el programa principal y éste es capaz de administrar todos sus recursos para ser utilizados de manera eficiente, cómoda y sin interrupciones, de tal manera que el usuario pueda mantener una comunicación sin problemas haciendo uso de los recursos que el hardware le suministra, (Malave Polanco & Beauperthuy Taibo, 2011).

Android es un sistema operativo con una plataforma abierta para dispositivos móviles adquirido por Google y la *Open Handset Alliance*, su finalidad es satisfacer la necesidad de

los operadores móviles y fabricantes de dispositivos, además de fomentar el desarrollo de aplicaciones, cualidad que ningún otro sistema operativo incluye en sus conceptos. Android bajo la definición de Google se considera un “*software stack*” o una pila de software, ya que está conformada por:

- El sistema operativo, donde todas las funciones se desarrollan.
- El middleware que permite la conexión entre redes.
- Las aplicaciones o API's que constituyen todos los programas que el teléfono puede ejecutar.

2.13 Características de Android

Este sistema operativo se torna realmente atractivo por diversas características, entre ellas se encuentran, (Malave Polanco & Beauperthuy Taibo, 2011):

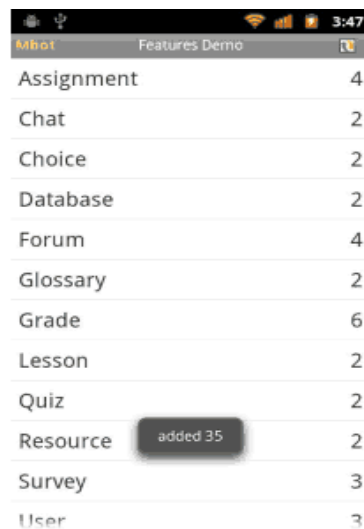
- Plataforma totalmente libre basado en Linux que permite desarrollar aplicaciones y/o modificar las ya existentes con lenguaje de Java.
- Es *multitasking* permitiendo mantener distintas aplicaciones corriendo al mismo tiempo.
- Compatible con una gran variedad de hardware en el mercado (tablets y dispositivos celulares de marcas como: Motorola, Samsung, ZTE, Huawei, Ericsson por nombrar algunas).
- Posee un portal llamado *Android Market* donde se tiene acceso a muchas aplicaciones que pueden ser utilizadas.
- Permite realizar actualizaciones del sistema operativo en línea siempre y cuando el dispositivo soporte los requerimientos del mismo.
- Puede operar soluciones tecnológicas referentes al uso de redes sociales, mensajería instantánea, correo electrónico, modificación y lectura de procesadores de palabras, hojas de cálculo, presentaciones, lectura de formatos pdf, entre otros.

2.14 Ejemplos de aplicaciones de *Android* a *Moodle*

1. Mbot (Android)

A pesar de que esta aplicación se referencia varias veces desde el sitio web de *Moodle*, no es fácil su localización ya que no se encuentra en el Market de Android (*Google Play*), ni tampoco en la web del desarrollador (*CodeGuild*), (Arjona Heredia & Gámiz Sanchez, 2013).

La aplicación muestra una lista en la que aparecen todos los tipos de recursos que es capaz de reconocer y el número de los mismos que encuentra en el curso, como se puede ver en la *Figura 3*. Esta organización resulta poco intuitiva ya que no refleja la estructura general del curso.



The screenshot shows a mobile application interface with a status bar at the top displaying the time 3:47 and various icons. Below the status bar, there is a header with the text 'Mbot' and 'Features Demo'. The main content is a list of resource types with their corresponding counts:

Resource Type	Count
Assignment	4
Chat	2
Choice	2
Database	2
Forum	4
Glossary	2
Grade	6
Lesson	2
Quiz	2
Resource	2
Survey	3
User	3

Figura 3. Pantalla de la aplicación

Del mismo modo, la aplicación utiliza también una lista similar a ésta para mostrar todos los recursos de un tipo, pero cuando se accede a uno de los recursos en concreto se abre una “vista web”, que muestra sin ninguna transformación ni adaptación la Web de *Moodle* tal y como se vería en un navegador normal de PC, pero en una pantalla mucho más pequeña. También surgen problemas a la hora de llevar a cabo operaciones complejas como, por ejemplo, la subida de archivos.

2. Doodle(Android)

Como ocurre con el caso anterior, esta es otra aplicación para *Android* que también necesitaría actualizarse en gran medida. En la *figura 4* se puede ver una captura de pantalla del aspecto que tiene la aplicación en funcionamiento. Su gran limitación radica en que solamente es capaz de mostrar las actividades de las versiones 2.1 y 2.2 de *Moodle*, sin dar soporte para el envío de las mismas.



Figura 4. Pantalla de la aplicación

Es necesario tener en cuenta que a partir de la versión 2.3 de *Moodle* se han introducido grandes cambios en el funcionamiento de las actividades, de manera que se pueden encontrar cursos en los que conviven dos tipos diferentes de actividades: las actividades en formato “antiguo” (a extinguir), que sí son soportadas por Doodle y las actividades en formato “nuevo” no soportadas por Doodle y que, por tanto, no aparecerían en la aplicación.

CAPÍTULO 3
MODELO CONCEPTUAL

3.1 Introducción

En este capítulo abordaremos la estructura del módulo de tareas de *moodle*, que es el apartado en el que trabajaremos para el desarrollo de nuestra aplicación, además de visualizar la estructura de este módulo a través de diagramas.

3.2 Las actividades de Moodle

En las acciones formativas a través de cualquier Entorno Virtual de Aprendizaje (EVA), es el mediador entre usuario y el EVA, el factor decisivo a la hora de establecer los procesos de aprendizaje, condicionando las actividades que se pueden llevar a cabo en el mismo. *Moodle* dispone de algunas herramientas o recursos orientados a posibilitar el aprendizaje desde el constructivismo social (Belloch, 2010).

En este sentido, las actividades de *Moodle* deben cubrir diferentes funciones en la acción formativa: **Aprendizaje**, cuando las actividades se orientan principalmente hacia la construcción de conocimiento ajustándose a diferentes modalidades (individual, grupal, colaborativa); **Evaluación**, si las tareas permiten la evaluación del alumno o facilitan instrumentos para el conocimiento del mismo; **Comunicación**, en él se engloban las actividades que la plataforma utilizará para facilitar la comunicación entre los participantes, tanto a nivel síncrono como asíncrono. Por tanto, *Moodle* dispone de los tipos de actividades que a continuación se mencionan.

Tabla 3. Muestra las actividades que existen en moodle

Aprendizaje	Evaluación	Comunicación
Recursos	Tareas	Foros
Glosarios	Cuestionarios	Diarios
Lección	Ejercicios	Consultas
Etiquetas	Talleres	Chat
	Encuestas	

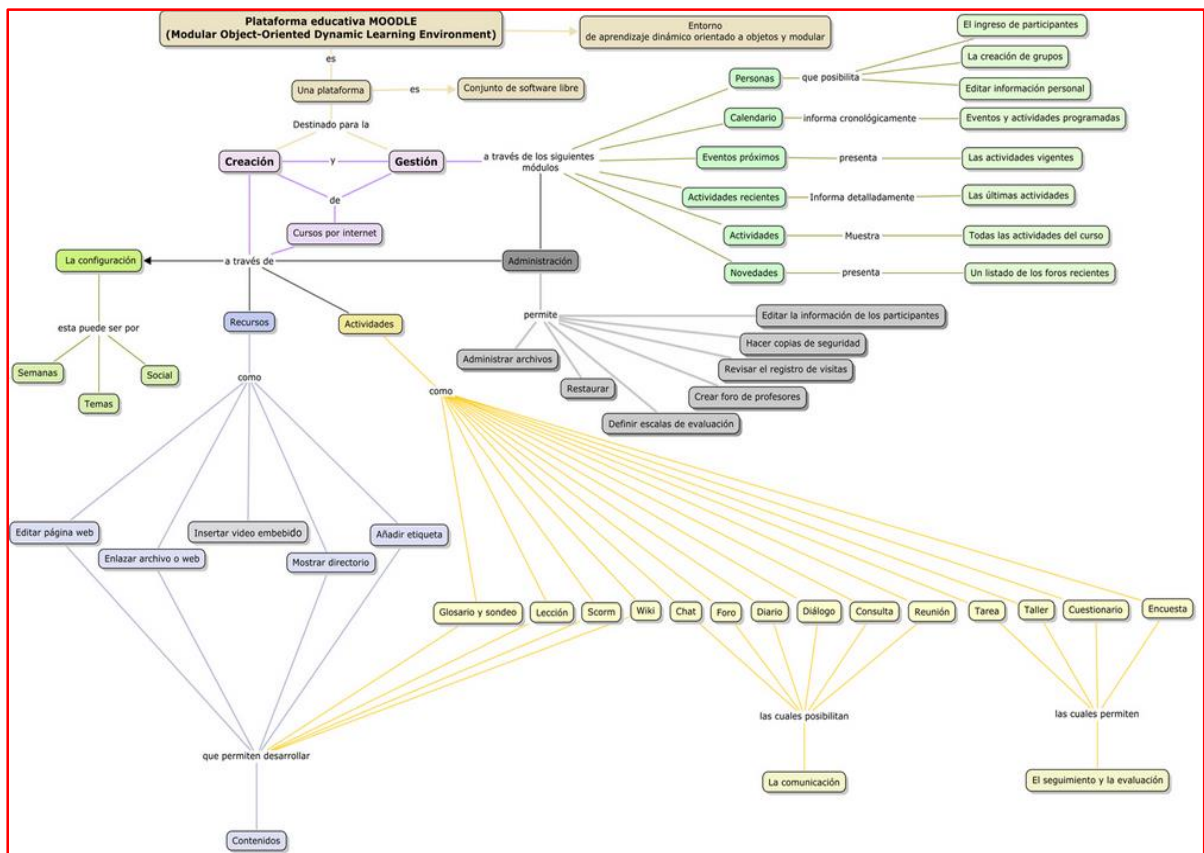


Figura 5. Visión General de los recursos de Moodle (Ospina P, 2008).

3.3 Estructura del módulo de Actividades o evaluación

Moodle dispone de una serie de actividades orientados al trabajo colaborativo, el módulo de actividades o evaluación, permite a un profesor evaluar el aprendizaje de los alumnos mediante un trabajo que posteriormente el revisará, valorará y calificará (Belloch, 2010).

Estas actividades pueden ser de tres tipos: una de carácter comunicativo, en ellas se engloban los medios de comunicación como los foros, diarios, diálogos, consultas y chat. Por otra parte pueden ser de forma de contenidos como los recursos, glosarios, lecciones y etiquetas. Y la tercera manera, es de tipo evaluativa, aquí entran los cuestionarios, talleres,

ejercicios, encuestas, calificaciones y tareas. Nosotros nos enfocaremos en este último recurso que es de carácter evaluativo (tareas).

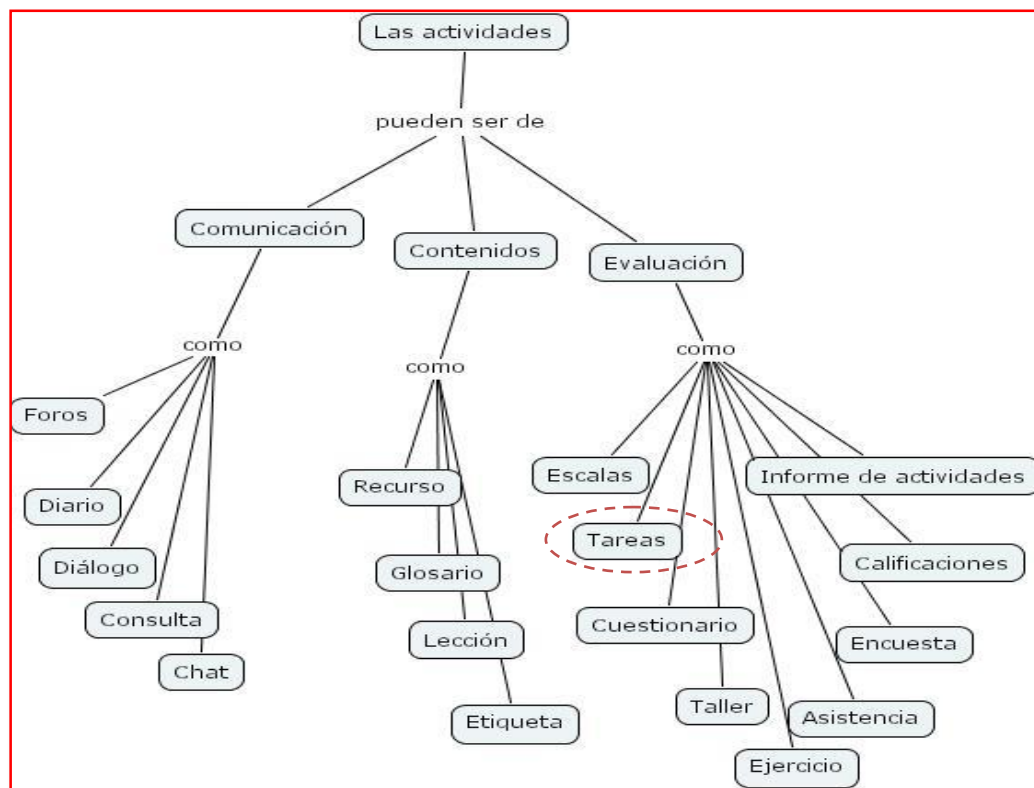


Figura 6. Muestra el módulo de las actividades de Moodle (Vicent, 2004).

3.4 Módulo de tareas

La parte en la que estaremos trabajando será en las actividades evaluativas, concretamente en el módulo de tareas, en este módulo, existen 4 tipos de tareas, en la cual el profesor marca una tarea y los alumnos son los que realizan dicha actividad en una fecha y tiempo determinado, estas tareas son las siguientes:

- Texto en línea
- Subir un solo archivo
- Uploading avanzada de archivos
- Actividades desconectadas.

En cada una de estas tareas, el propósito es que el alumno suba un trabajo en la fecha y hora que el profesor establezca, además de cumplir con los requisitos que el mentor solicite para el trabajo.

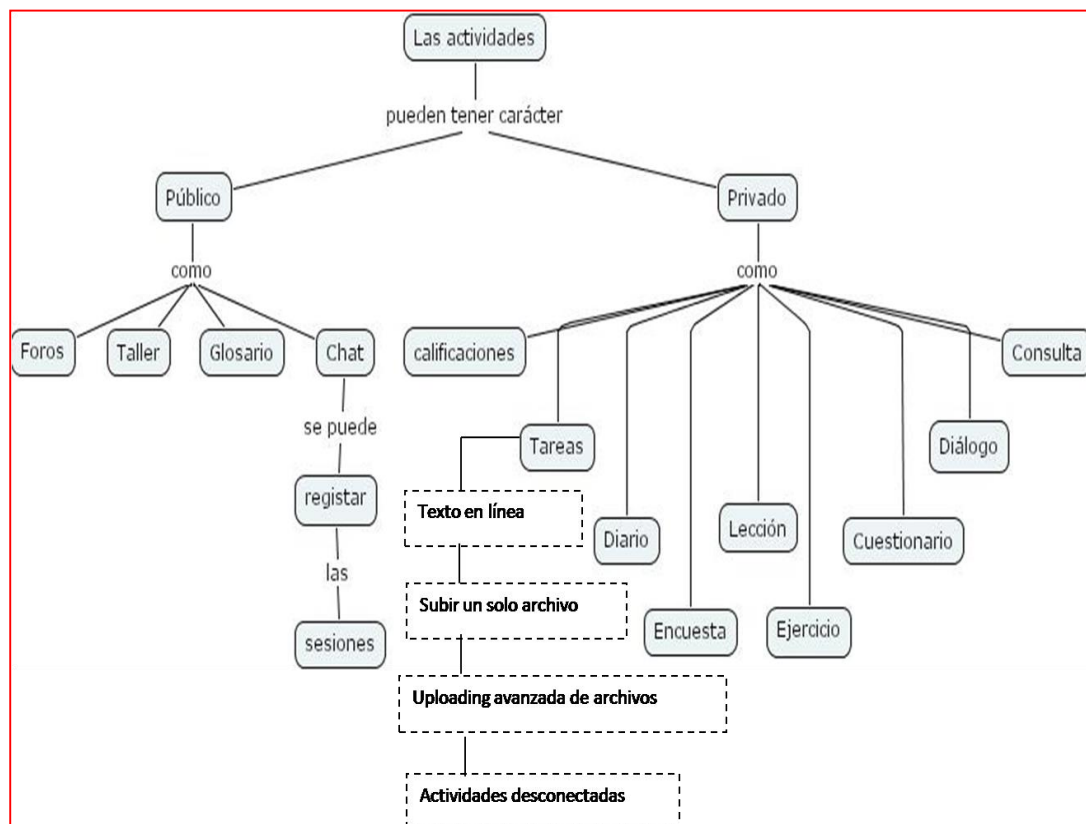


Figura 7. Muestra los recursos del módulo de actividades tareas (Vicent, 2004).

3.4.1 Subir un solo archivo

Un estudiante puede cargar un único archivo, como un documento de Word, hoja de cálculo o cualquier cosa digital. Varios archivos pueden ser comprimidos y luego presentarse. Después los alumnos suben sus archivos, el profesor será capaz de abrir la presentación y luego usar la interfaz de Moodle para asignar una calificación y ofrecer comentarios como retroalimentación (Baño Sancho, 2007).

3.4.2 Uploading avanzada de archivos

Las opciones incluyen la presentación de múltiples archivos, permitiendo a los estudiantes para escribir un mensaje junto a su presentación y que regrese un archivo como retroalimentación.

3.4.3 Texto en línea

Este tipo de tarea solicita a los usuarios componer y editar texto, utilizando las herramientas de edición normales. La asignación de texto en línea se puede configurar para permitir a los estudiantes componer, revisar y editar a través del tiempo o de modo que el alumno sólo tiene una oportunidad de visualizar su respuesta. Por otra parte, con la asignación en línea, los profesores pueden calificar el trabajo en línea e incluso editar y/o formular comentarios en el trabajo del alumno. La asignación de texto en línea es ideal para meter en diario y la composición del trabajo.

3.4.4 Actividad desconectada

Esto es útil cuando la tarea se realiza fuera de Moodle. Podría ser algo en otro lugar en la web o cara a cara. Los estudiantes pueden ver una descripción de la tarea, pero no pueden subir archivos. La calificación funciona normalmente y los estudiantes recibirán

notificaciones de sus calificaciones. En concreto, un profesor en un entorno mixto (combinación de cara a cara y la instrucción en línea) puede utilizar el tipo de actividad fuera de línea de asignación para asignar un proyecto que el alumno se presente físicamente al profesor.

3.5 Modelo conceptual de la actividad (Tareas)

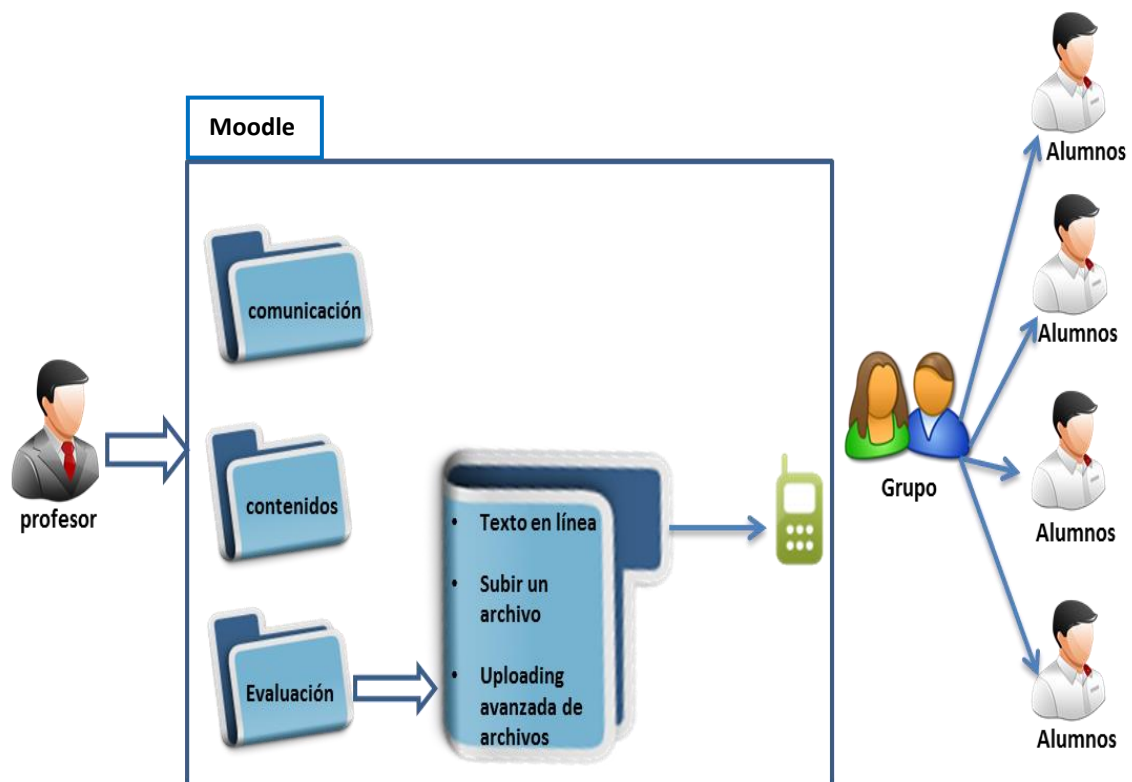


Figura 8. Muestra el Diagrama conceptual del módulo de tareas.

En este modelo conceptual, visualizamos al profesor el cual tiene para marcas tres tipos de actividades o tareas pero en el módulo que trabajaremos para lo de las notificaciones será en el de evaluación, el cual consta de tres tareas y/o actividades, las cuales al ser marcadas por el mentor y al darle clic para subirlo esta generará un aviso de notificaciones y serán enviados a los celulares de los alumnos que estén registrados en ese curso o grupo y los cuales cuentan con celulares o dispositivos con sistemas operativos *Android*.

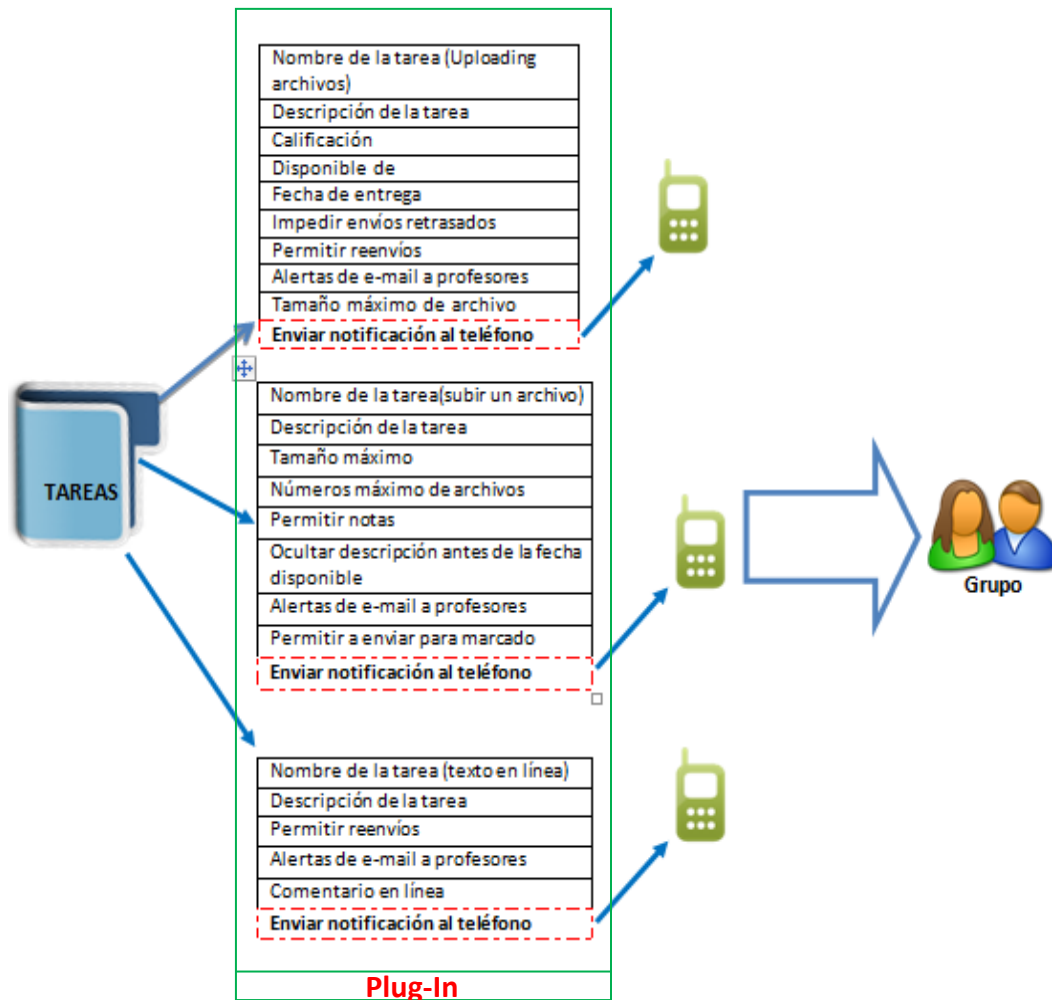


Figura 9. Muestra los datos de cada tarea.

En este modelo conceptual mostramos las tres tareas principales del módulo de evaluación, las cuales al ser subidas generaran las notificaciones que serán enviadas a los usuarios que estén registrados en el grupo o materia y que cuentan con dispositivos *Android*.

Cabe mencionar que únicamente se recibirá el aviso de las notificaciones cuando se suban cualquiera de estas tres tareas (Uploading avanzados de archivos, subir un solo archivo y texto en línea) y al darle clic al botón subir este generará el aviso a aquellas personas que estén registradas o matriculadas solamente en el curso.

CAPÍTULO 4
IMPLEMENTACIÓN Y RESULTADOS

4.1 Introducción

En este capítulo abordaremos todo lo referente a la implementación de nuestro proyecto, desde la construcción de la aplicación para el dispositivo *Android*, los diagramas de casos de usos del módulo, diagramas de robustez, hasta las modificaciones que se realizaron en *moodle* para las notificaciones. Además de visualizar las clases que interactúan para dicho evento y ver cuáles fueron las que se modificaron. Y por último los resultados de nuestra proyecto.

4.2 Modelo de clases del módulo de tareas

En la **figura 10** podemos visualizar el diagrama de clases para el módulo de tarea, en donde podemos identificar cada una de las tareas respectivamente con sus atributos y al tipo de datos que pertenecen cada uno de ellos. Además de visualizar el envío de la notificación que tendrá cada una de las tareas y que ésta se generará cuando se dé clic en el botón de guardar, para subir la tarea. Nosotros como usuarios docentes, dentro de *moodle* podemos elegir cualquier actividad que vayamos a usar para los alumnos, dentro de ellas una es la tarea, existen tres tipos de tareas las cuales ya vimos en el capítulo anterior, con las cuales podemos subirlas, modificarlas cuando ya se subieron o simplemente, eliminarlas si así queremos.

Las clases que se usan para el módulo de tarea son las siguientes, una clase llamada *Assigment_base* la cual contiene los métodos/opciones que se pueden hacer en ella como (*Add_instance*, *Update_instance* y *Delete_instance*), las cuales de allí se divide para los tres tipos de tareas (*assigment_upload*, *assigment_upload single* y *assigment_online*). Los cuales contienen atributos que heredan de la clase *Assigment_base*.

A continuación veremos un pequeño diagrama de clases, de lo mencionado anteriormente en donde podemos ver las clases y cuáles son los atributos que contiene cada una de ellas, además de visualizar el apartado que se modificó para el envío de los avisos a los dispositivos *Android*. Esto fue en cada una de las clases de las respectivas tareas en los cuales se le modifico el evento del botón, para poder hacer el envío de las notificaciones.

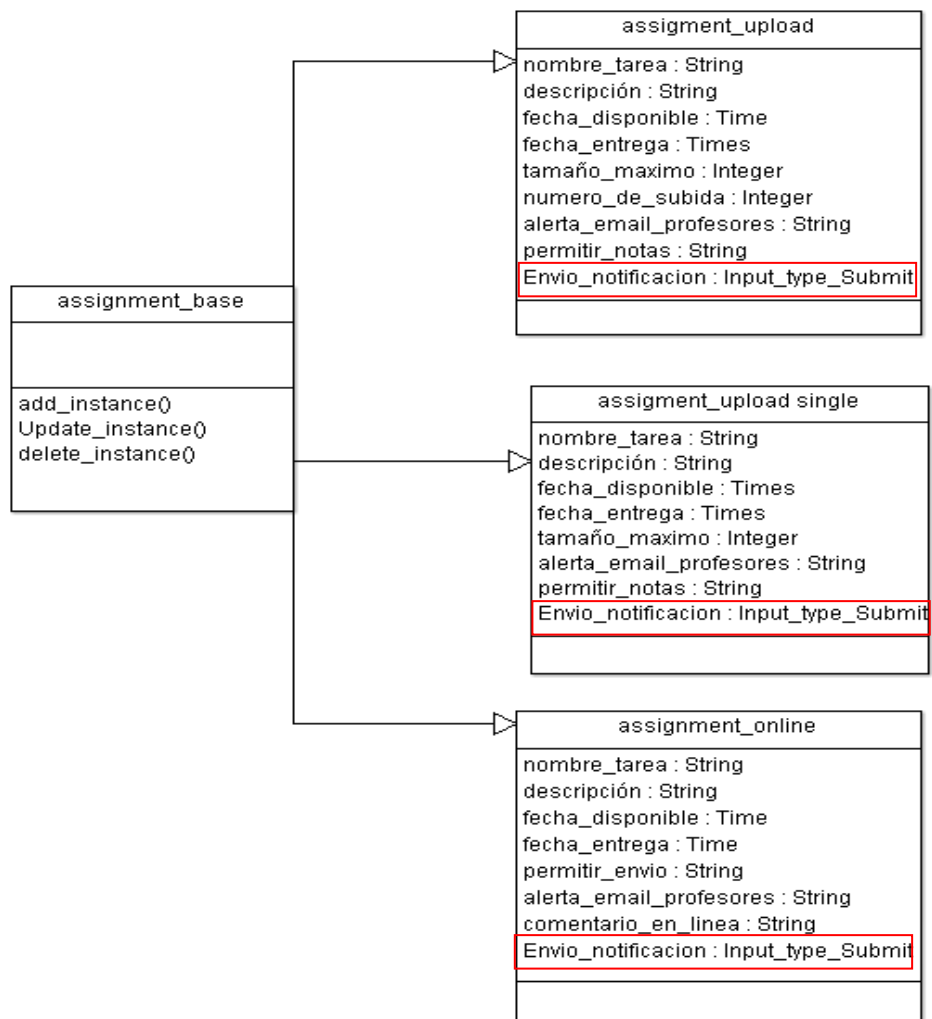


Figura 10. Muestra el diagrama de clases y el atributo de las tareas.

4.3 Diagramas de Casos del módulo de (Tareas)

En este diagrama podemos ver el módulo de tarea de *moodle* que se va a modificar, en donde encontramos que el profesor puede crear, editar y eliminar una tarea. Y el alumno visualizará esa tarea subida por el mentor.

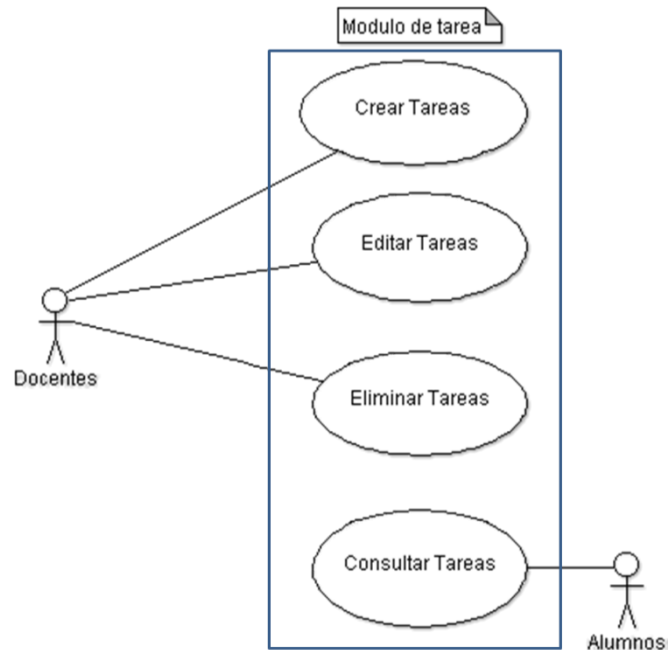


Figura 11. Casos de uso relacionado con la tarea

Tabla 4. Muestra cuando se crea una tarea

Nombre del caso de uso	
Crear Tarea	
Descripción:	
El usuario docente podrá crear y añadir a un curso una tarea. De las tres tareas antes mencionadas.	
Para ello deberá:	
<ul style="list-style-type: none"> Introducir un nombre a la tarea (obligatorio) y rellenar los demás campos del formulario. 	

Tabla 5. *Muestra cuando se edita una tarea*

Nombre del caso de uso	
Editar Tarea	
Descripción:	
<ul style="list-style-type: none">• El usuario docente podrá cambiar las propiedades de las tareas creadas con anterioridad.	

Tabla 6. *Muestra cuando se elimina una tarea*

Nombre del caso de uso	
Eliminar Tarea	
Descripción:	
<ul style="list-style-type: none">• El usuario docente podrá eliminar las tareas creadas con anterioridad.	

Tabla 7. *Muestra cuando se consulta una tarea*

Nombre del caso de uso	
Consultar Tarea	
Descripción:	
<ul style="list-style-type: none">• El usuario alumno podrá consultar la tarea en la plataforma, pero antes ya fue notificado a través del dispositivo <i>Android</i> que existe una tarea que el docente subió.	

4.4 Diagramas de robustez, para subir (Tareas)

En estos diagramas podemos visualizar los pasos que el sistema realiza a la hora de que el profesor crea, edita y elimina, unas de las tres tareas, las cuales realizarán el envío de las notificaciones a los dispositivos de los alumnos.

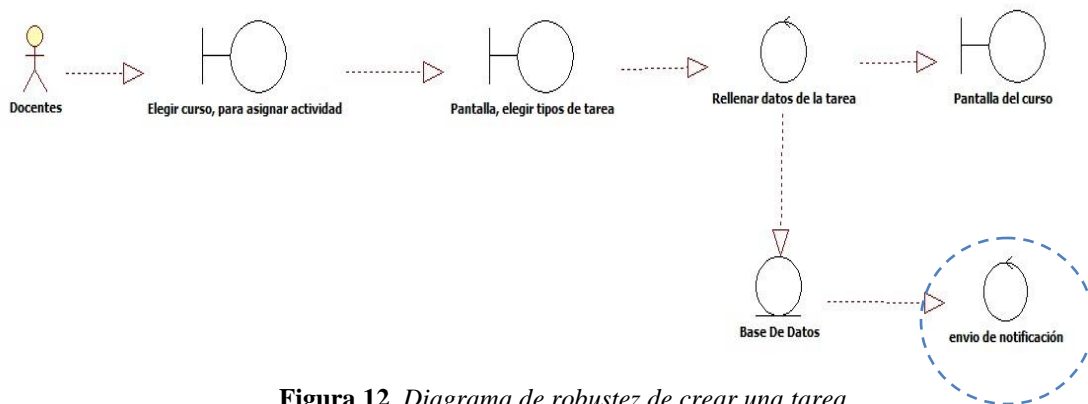


Figura 12. Diagrama de robustez de crear una tarea

Tabla 8. Muestra las características cuando se sube una tarea

Subir o crear una tarea	
Precondiciones	El usuario ha de estar identificado en moodle como docente. El usuario ha de encontrarse dentro del curso en el que quiera crear o subir la tarea.
Poscondiciones	Se crea una nueva tarea que se guarda en la base de datos y se envía la notificación a los alumnos que están en ese curso.
Actores	Usuario docente
Descripción	<ol style="list-style-type: none"> 1. La plataforma moodle muestra la página principal en la que el docente elegirá el tipo de tarea a subir. 2. El usuario introduce el nombre de la tarea (obligatorio ya que es el que se mandará en la notificación), y los demás datos del formulario de la tarea. 3. El usuario guarda la tarea dándole clic en cualquiera de los dos botones que aparecen al final del formulario, los cuales realizan el envío del mensaje a los dispositivos. 4. Moodle valida, guarda y realiza el envío de los mensajes a los alumnos que están en el curso.

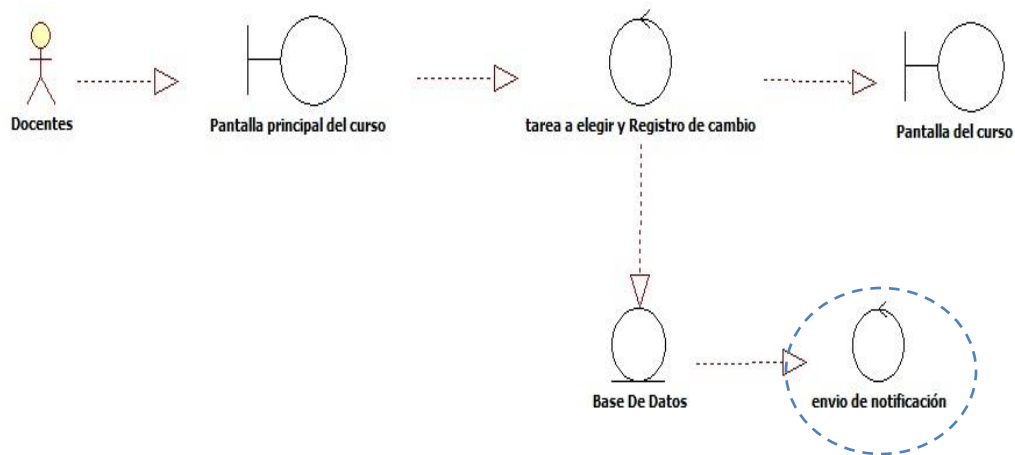


Figura 13. Diagrama de robustez de editar una tarea

Tabla 9. Muestra las características cuando se edita una tarea

Editar una tarea	
Precondiciones	El usuario ha de estar identificado en moodle como docente. La tarea a modificar ha de estar en la base de datos.
Poscondiciones	Se guardan los cambios de la tarea en la base de datos y se envía nuevamente la notificación.
Actores	Usuario docente
Descripción	<ol style="list-style-type: none"> 1. Moodle muestra el formulario de edición de la tarea. 2. El usuario introduce los cambios que desee. 3. Moodle valida la nueva información, guarda y realiza nuevamente el envío de la notificación al alumno.

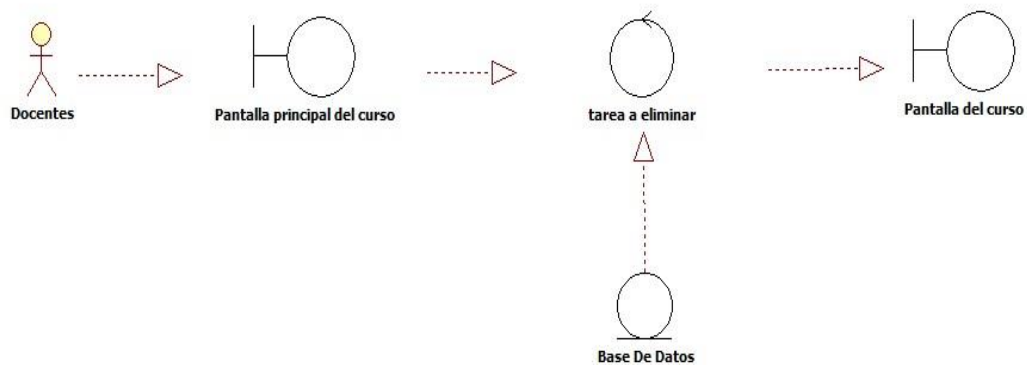


Figura 14. Diagrama de robustez de eliminar una tarea

Tabla 10. Muestra las características cuando se elimina una tarea

Eliminar una tarea	
Precondiciones	El usuario ha de estar identificado en moodle como docente. La tarea a eliminar ha de estar en la base de datos.
Poscondiciones	Se elimina la tarea de la base de datos.
Actores	Usuario docente
Descripción	<ol style="list-style-type: none"> 1. Moodle muestra todas las tareas del curso. 2. El usuario selecciona la tarea que quiere eliminar. 3. El sistema borra la tarea de la base de datos.

4.5 Diagramas de interacción, para el módulo de (Tareas)

En estos diagramas se muestra la interacción que tiene el usuario con el sistema (*moodle*), además de ver cuáles son las clases con la que se interactúa, a la hora de realizar el envío de las notificaciones a los dispositivos de los alumnos.

En este primer diagrama se puede ver las clases que son usadas, cuando se sube o se crea unas de las tres tareas que existe en *moodle* y las cuales realizan el envío de una notificación a los alumnos que están en ese grupo.

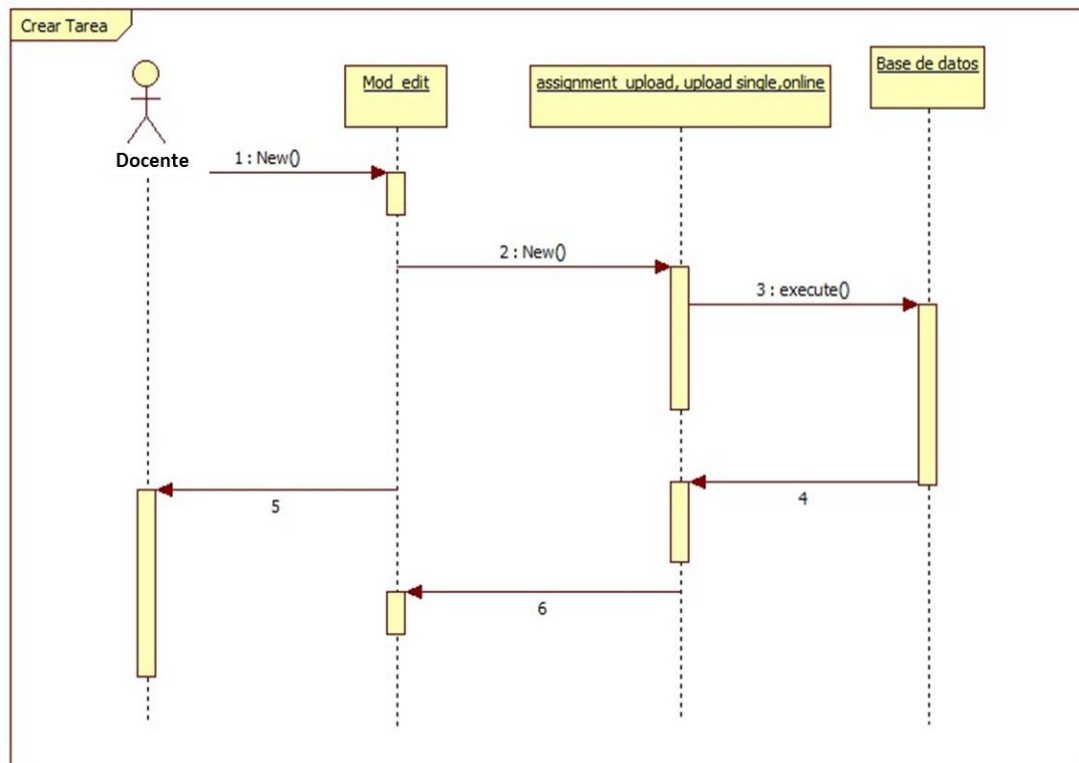


Figura 15. Diagrama de secuencia para crear una tarea

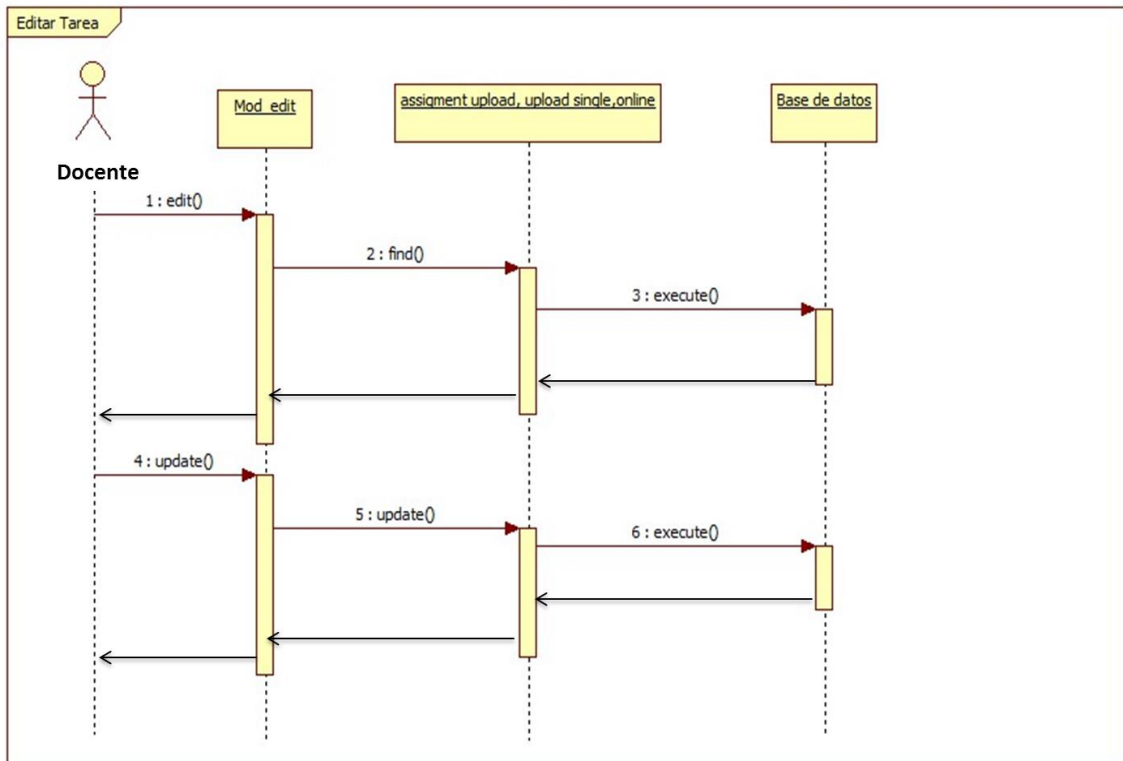


Figura 16. Diagrama de secuencia para editar una tarea

En este segundo diagrama visualizamos, cuando el docente realiza algún cambio en la tarea, cuáles son las clases que se modifican o actúan en este proceso.

El docente primero debe entrar a la plataforma y al curso en donde quiere modificar la tarea, después elige la tarea a modificar y realiza los cambios a ella y lo guarda, el sistema realiza el guardado y envío de la notificación a los alumnos inscritos en ese curso.

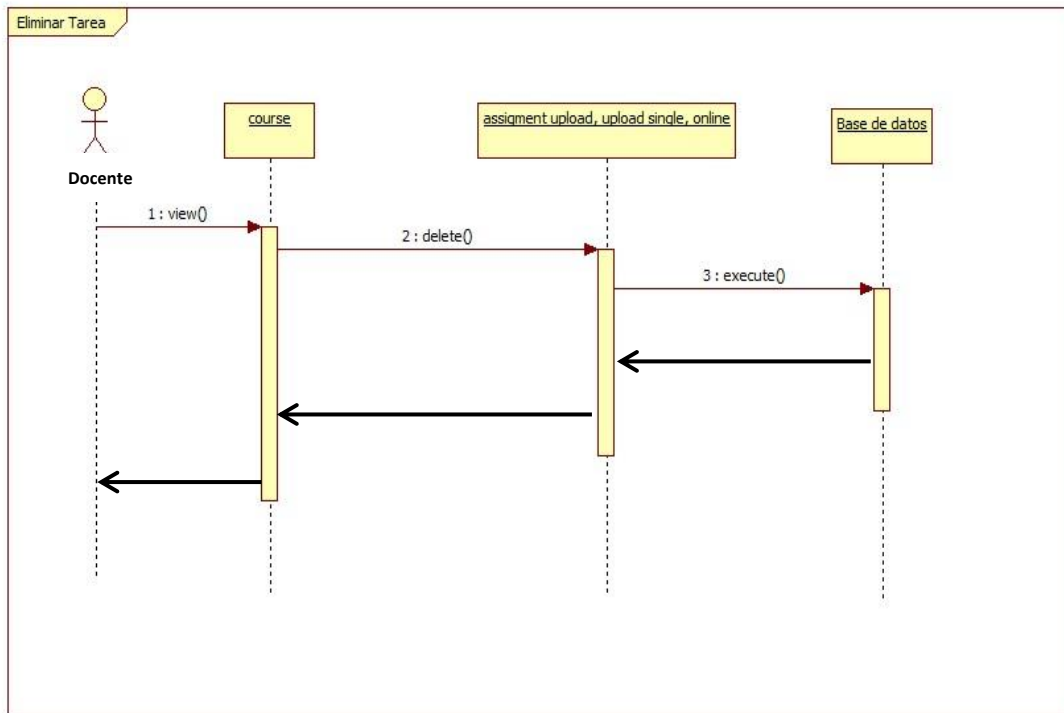


Figura 17.Diagrama de secuencia para eliminar una tarea

En este diagrama podemos ver cuando el usuario docente quiere eliminar una tarea específica, primero debe entrar al curso de donde quiere eliminar la tarea subida, posteriormente elige la tarea a borrar.

El sistema le preguntará si realmente quiere eliminarla al darle clic elimina la tarea y la borra de la base datos por lo cual el alumno ya no podrá ver la tarea en la plataforma.

4.6 Implementación de las interfaces de Android

Una de las partes más importantes a la hora de realizar un sistema o aplicación, sin duda alguna es la presentación o la interfaz, que básicamente es la parte con la que el usuario interactúa directamente, nosotros desarrollamos la aplicación de Android en el programa de Eclipse Classic en su versión V3.3.1 usando el instalador para Android SDK.

A continuación veremos el proceso de desarrollo de cada una de las pantallas para el programa que será instalado en el dispositivo, además de la utilización de PHP como lenguaje de programación del lado del servidor y MySQL como base de datos del servidor para poder recibir las notificaciones por parte de la plataforma *moodle*.

4.6.1 Creación del proyecto Android y registro en Google Cloud Mensajería (GCM)

Como primer paso para poder realizar este proyecto, tuvimos que registrarnos en la página de *Google Cloud Mensajería* desde GOOGLE consola de API y obtener Sender ID y clave de API de Google Cloud Mensajería. Después Creamos un proyecto Android para registrarse en Google Cloud Mensajería (GCM).

Posteriormente se procedió a crear un proyecto nuevo en eclipse, en la cual se crearon las clases que se usarían en el proyecto y que más adelante se describirán cada una de ellas. El proyecto cuenta con 8 clases, las cuales se explicarán las más importantes a groso modo.

Tabla 11. Muestra las clases del proyecto de Android.

AlertDialogManager.java
CommonUtilities.java
ConnectionDetector.java
GcmIntentService.java
MainActivity.java
RegisterActivity.java
ServerUtilities.java
WakeLocker.java

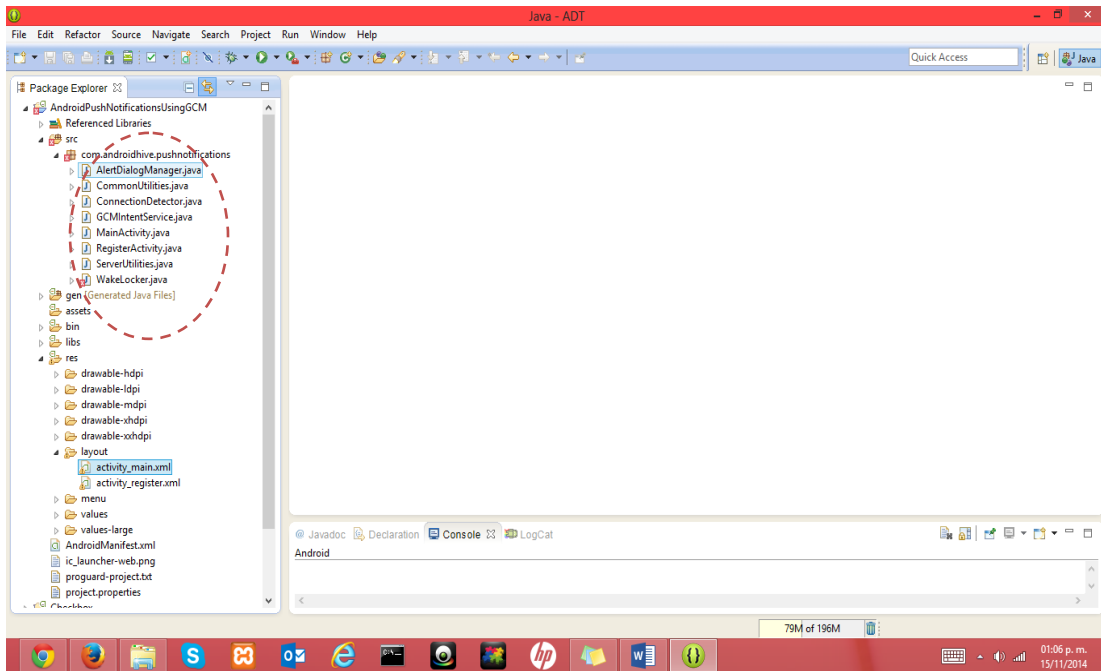


Figura 18. Muestra la aplicación en eclipse y sus clases

AlertDialogManager.java

Esta clase se utiliza para mostrar diálogo de alerta en su aplicación.

```
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;

public class AlertDialogManager {

    public void showAlertDialog(Context context, String title, String message,
        Boolean status) {
        AlertDialog alertDialog = new AlertDialog.Builder(context).create();

        // Setting Dialog Title
        alertDialog.setTitle(title);

        // Setting Dialog Message
        alertDialog.setMessage(message);
        if(status != null)
            // Setting alert dialog icon
            alertDialog.setIcon((status) ? R.drawable.success : R.drawable.fail);

        // Setting OK Button
        alertDialog.setButton("OK", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
            }
        });
        // Showing Alert Message
        alertDialog.show();
    }
}
```

Figura 19. Muestra el código de la clase "AlertDialogManager.java"

CommonUtilities.java

Esta clase contiene la configuración GCM y nuestra URL de registro del servidor. SERVER_URL - El servidor de registro de usuario URL SENDER_ID - Identificación del proyecto Google.

```
package com.androidhive.pushnotifications;

import android.content.Context;
import android.content.Intent;

public final class CommonUtilities {

    // give your server registration url here
    static final String SERVER_URL="http://192.168.1.140/gcm_server_php/register.php";

    // Google project id
    static final String SENDER_ID = "699806067594";

    /**
     * Tag used on log messages.
     */
    static final String TAG = "AndroidHive GCM";

    static final String DISPLAY_MESSAGE_ACTION =
        "com.androidhive.pushnotifications.DISPLAY_MESSAGE";

    static final String EXTRA_MESSAGE = "message";
}
```

Figura 20. Muestra el código de la clase "CommonUtilities.java"

ConnectionDetector.java

Esta clase se utiliza para detectar el estado de conexión a Internet.

```
import android.content.Context;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;

public class ConnectionDetector {

    private Context _context;

    public ConnectionDetector(Context context){
        this._context = context;
    }

    /**
     * Checking for all possible internet providers
     * **/
    public boolean isConnectingToInternet(){
        ConnectivityManager connectivity = (ConnectivityManager)
        _context.getSystemService(Context.CONNECTIVITY_SERVICE);
        if (connectivity != null)
        {
            NetworkInfo[] info = connectivity.getAllNetworkInfo();
            if (info != null)
            {
                for (NetworkInfo ni : info)
                {
                    if (ni.isConnected())
                    {
                        return true;
                    }
                }
            }
        }
        return false;
    }
}
```

Figura 21. Muestra el código de la clase "ConnectionDetector.java"

GcmIntentService.java

Esta clase se encarga de todos los servicios relacionados con el GCM.

```
import static com.androidhive.pushnotifications.CommonUtilities.SENDER_ID;
import static com.androidhive.pushnotifications.CommonUtilities.displayMessage;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.util.Log;

import com.google.android.gcm.GCMBaseIntentService;

public class GCMIntentService extends GCMBaseIntentService {

    private static final String TAG = "GCMIntentService";

    public GCMIntentService() {
        super(SENDER_ID);
    }

    /**
     * Method called on device registered
     */
    @Override
    protected void onRegistered(Context context, String registrationId) {
        Log.i(TAG, "Device registered: regId = " + registrationId);
        displayMessage(context, "Your device registered with GCM");
        Log.d("NAME", MainActivity.name);
        ServerUtilities.register(context, MainActivity.name,
            MainActivity.email, MainActivity.ID_curso, registrationId);
    }

    /**
     * Method called on device un registred
     */
    @Override
    protected void onUnregistered(Context context, String registrationId) {
        Log.i(TAG, "Device unregistered");
        displayMessage(context, getString(R.string.gcm_unregistered));
        ServerUtilities.unregister(context, registrationId);
    }

    /**
     * Method called on Receiving a new message
     */
    @Override
    protected void onMessage(Context context, Intent intent) {
        Log.i(TAG, "Received message");
        String message = intent.getExtras().getString("price");

        displayMessage(context, message);
        // notifies user
        generateNotification(context, message);
    }
}
```

Figura 22. Muestra el código de la clase "GCMIntentService.java"

ServerUtilities.java

Esta clase tiene la función siguiente

- Función para registrar un usuario en nuestro servidor (nombre, correo electrónico, ID curso GCM Identificación del reg.)
- Método para enviar datos al servidor

```
static void register(final Context context, String name, String
email,String ID_curso, final String regId) {
    Log.i(TAG, "registering device (regId = " + regId + ")");
    String serverUrl = SERVER_URL;
    Map<String, String> params = new HashMap<String, String>();
    params.put("regId", regId);
    params.put("name", name);
    params.put("email", email);
    params.put("ID_curso", ID_curso);

    long backoff = BACKOFF_MILLI_SECONDS + random.nextInt(1000);
    // Once GCM returns a registration id, we need to register on
our server
    // As the server might be down, we will retry it a couple
// times.
    for (int i = 1; i <= MAX_ATTEMPTS; i++) {
        Log.d(TAG, "Attempt #" + i + " to register");
        try {
            displayMessage(context, context.getString(
                R.string.server_registering, i, MAX_ATTEMPTS));
            post(serverUrl, params);
            GCMRegistrar.setRegisteredOnServer(context, true);
            String message =
context.getString(R.string.server_registered);
                CommonUtilities.displayMessage(context, message);
            return;
        } catch (IOException e) {
            // Here we are simplifying and retrying on any error;
in a real
            // application, it should retry only on unrecoverable
errors
            // (like HTTP error code 503).
            Log.e(TAG, "Failed to register on attempt " + i + ":" +
e);

            if (i == MAX_ATTEMPTS) {

                String message =
context.getString(R.string.server_register_error,
                    MAX_ATTEMPTS);
                CommonUtilities.displayMessage(context, message);
            }
            break;
        }
    }
}
```

Figura 23. Muestra el código de "ServerUtilities.java"

4.6.2 Desarrollo de las pantallas en Android (eclipse)

En la **figura 24**. Podemos ver la primera pantalla de nuestra aplicación de Android, la cual contiene 3 *TextView* (uno para el nombre del usuario, para el email y para el Id curso) de igual manera contiene 3 *EditText* para que el usuario pueda ingresar los datos que se piden. Y son guardadas en la base de datos de *moodle*.

Y un botón llamado registrar para mandar los datos al servidor de GCM de *google*.

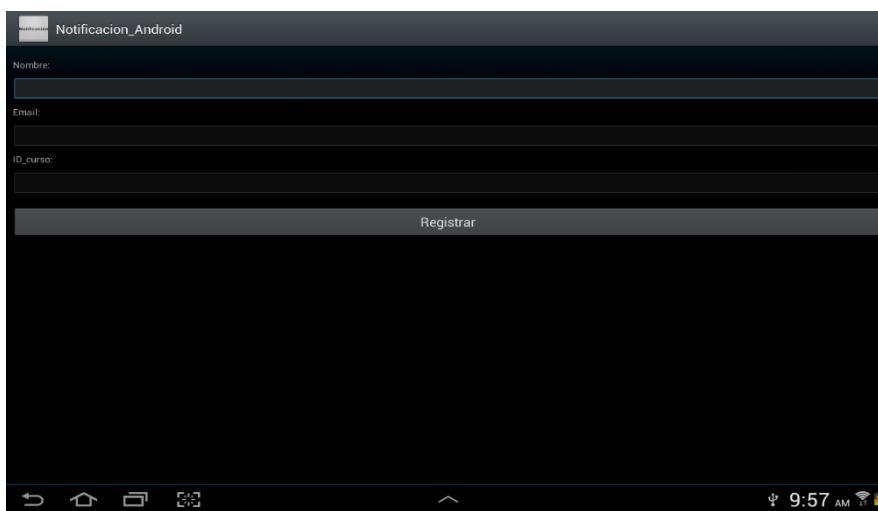


Figura 24. Pantalla principal de la aplicación en Android

En la **figura 25**. Se visualiza el código de la primera pantalla de Android, la cual se describió con todos los componentes que esta contiene en la hoja anterior.

```
<TextView android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:text="Nombre:"
  android:layout_marginLeft="10dip"
  android:layout_marginRight="10dip"
  android:layout_marginTop="20dip"/>

<EditText android:id="@+id/txtName"
  android:layout_width="fill_parent"
  android:layout_height="wrap_content"
  android:layout_margin="10dip"
  android:layout_marginBottom="20dip"/>

<TextView android:layout_width="fill_parent"
  android:layout_height="wrap_content"
```

Figura 25. Código que muestra la primera pantalla de la aplicación Android.

4.6.3 Pantalla de conexión al servidor

En la **figura 26**. Podemos ver la segunda pantalla de la aplicación, la cual aparece cuando ya ha sido rellenado los campos en la primera pantalla y se dio clic al botón registrar. Por consiguiente esos datos fueron enviados a la base datos de moodle y se pidió la conexión al servidor de GCM de google.

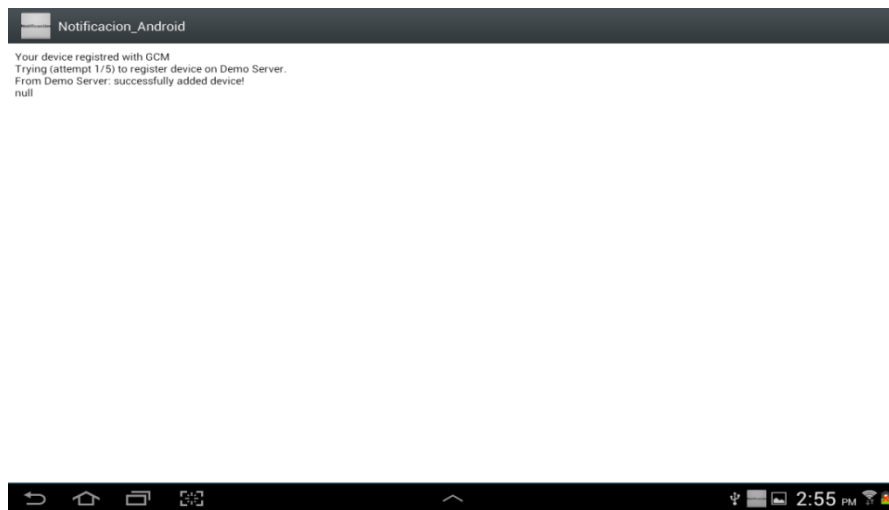


Figura 26.segunda pantalla de la aplicación en Android

En la **figura 27** se visualiza el código de la pantalla anterior, la cual cuenta con un TextView llamado (lblmessage), el cual sirve para poder mostrar el mensaje en pantalla, cuando este fue enviado por la plataforma al dispositivo.

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
  xmlns:tools="http://schemas.android.com/tools"
  android:layout_width="match_parent"
  android:layout_height="match_parent"
  android:orientation="vertical"
  android:background="#ffffff" >

  <TextView android:id="@+id/lblMessage"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:textSize="16dip"
    android:layout_margin="10dip"
    android:textColor="#000000"/>

</LinearLayout>
```

Figura 27.Muestra el Código de la pantalla de conexión al servidor

4.7 Creación de la tabla en la base de datos (moodle)

Para nuestro proyecto creamos dentro de moodle una tabla que se llama “*mdl_notificaciones_movil*”, en donde se insertaran los datos que mandara la aplicación. Con los datos (nombre, email e ID_curso).

```
CREATE TABLE IF NOT EXISTS `mdl_notificaciones_movil` (`id` int(11) NOT NULL AUTO_INCREMENT, `gcm_regid` text, `name` varchar(50) NOT NULL, `email` varchar(255) NOT NULL, `ID_curso` varchar(50) NOT NULL, `created_at` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP, PRIMARY KEY (`id`)) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1;
```

Figura 28. Muestra la sentencia para crear la tabla en la base de datos moodle

4.8 Creación y ejecución del proyecto PHP

Ahora veremos el código del lado del servidor para salvar Google Cloud Mensajería identificación del registro en nuestra base de datos y enviar notificación de inserción a dispositivo. El código se realizó en el lenguaje PHP.

A continuación mostraremos el primer código que lo que hace es la configuración hacia la base datos a conectar, en este caso es a la base datos de *moodle*. Contiene la clave de API de Google.

```
<?php
/**
 * Database config variables
 */
define("DB_HOST", "localhost");
define("DB_USER", "root");
define("DB_PASSWORD", "");
define("DB_DATABASE", "moodle");

/*
 * Google Cloud Messaging API Key
 */
define("GOOGLE_API_KEY", "AIzaSyB4d-Fh1qyPmx fzTaK-
BNTKzi0FhniBIqo"); // Place your Google API Key

?>
```

Figura 29. Muestra la configuración hacia la base datos moodle

db_connecte.php

Después de la configuración para conectarte con la base de datos, tenemos otro archivo hecho en PHP, llamado “**db_connecte.php**”, el cual básicamente se encarga de las conexiones de base de datos, se abre y se cierra la conexión principalmente.

```
<?php
class DB_Connect {
    // constructor
    function __construct() {
    }
    // destructor
    function __destruct() {
        // $this->close();
    }
    // Connecting to database
    public function connect() {
        require_once 'config.php';
        // connecting to mysql
        $con = mysql_connect(DB_HOST, DB_USER, DB_PASSWORD);
        // selecting database
        mysql_select_db(DB_DATABASE);
    }
    // Closing database connection
    public function close() {
        mysql_close();
    }
}
?>
```

Figura 30. Código para las conexiones de base de datos

db_functions.php

Este archivo contiene funciones para escribir la función, para la creación de usuario único.

```
<?php
class DB_Functions {

    private $db;

    //put your code here
    // constructor
    function __construct() {
        include_once './db_connect.php';
        // connecting to database
        $this->db = new DB_Connect();
        $this->db->connect();
    }

    // destructor
    function __destruct() {

    }

    /**
     * Storing new user
     * returns user details
     */
    public function storeUser($name, $email,$ID_curso,$gcm_regid) {
        // insert user into database
        $result = mysql_query("INSERT INTO mdl_notificaciones_movil(name,
email,ID_curso,gcm_regid, created_at) VALUES('$name',
'$email','$ID_curso','$gcm_regid', NOW())");
        // check for successful store
        if ($result) {
            // get user details
            $id = mysql_insert_id(); // last inserted id
            $result = mysql_query("SELECT * FROM mdl_notificaciones_movil
WHERE id = $id") or die(mysql_error());
            // return user details
            if (mysql_num_rows($result) > 0) {
                return mysql_fetch_array($result);
            } else {
                return false;
            }
        } else {
            return false;
        }
    }
}
```

Figura 31. Código para la creación de usuario único

GCM.php

Este archivo se utiliza para enviar solicitudes de notificación de inserción a servidor GCM.

```
public function send_notification($registatoin_ids, $message) {
    // include config
    include_once './config.php';

    // Set POST variables
    $url = 'https://android.googleapis.com/gcm/send';

    $fields = array(
        'registration_ids' => $registatoin_ids,
        'data' => $message,
    );

    $headers = array(
        'Authorization: key='. GOOGLE_API_KEY,
        'Content-Type: application/json'
    );
    // Open connection
    $ch = curl_init();

    // Set the url, number of POST vars, POST data
    curl_setopt($ch, CURLOPT_URL, $url);

    curl_setopt($ch, CURLOPT_POST, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, $headers);
    curl_setopt($ch, CURLOPT_RETURNTRANSFER, true);

    // Disabling SSL Certificate support temporarily
    curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);

    curl_setopt($ch, CURLOPT_POSTFIELDS, json_encode($fields));

    // Execute post
    $result = curl_exec($ch);
    if ($result === FALSE) {
        die('Curl failed: ' . curl_error($ch));
    }

    // Close connection
    curl_close($ch);
    echo $result;
}

?>
```

Figura 32. Código para enviar solicitudes de notificación de inserción a servidor GCM.

Register.php

Este archivo recibe peticiones de dispositivo *Android* y almacena el usuario en la base de datos.

```
<?php
// response json
$json = array();

/**
 * Registering a user device
 * Store reg id in users table
 */
if (isset($_POST["name"]) && isset($_POST["email"]) &&
isset($_POST["ID_curso"]) && isset($_POST["regId"])) {
    $name = $_POST["name"];
    $email = $_POST["email"];
    $ID_curso = $_POST["ID_curso"];
    $gcm_regid = $_POST["regId"]; // GCM Registration ID
    // Store user details in db
    include_once './db_functions.php';
    include_once './GCM.php';

    $db = new DB_Functions();
    $gcm = new GCM();

    $res = $db->storeUser($name, $email,$ID_curso, $gcm_regid);

    $registatoin_ids = array($gcm_regid);
    $message = array("product" => "shirt");

    $result = $gcm->send_notification($registatoin_ids, $message);

    echo $result;
} else {
    // user details missing
}
?>
```

Figura 33. Código para almacenar el usuario a la base datos

4.9 Implementación o modificación en moodle

Ahora mostraremos en este punto las configuraciones y/o modificaciones que realizamos en la plataforma *moodle*, para hacer que se mandaran las notificaciones de las tareas a los alumnos en un grupo específico.

Primer paso:

Moodle cuenta con varios módulos como se vio en el capítulo 3 del modelo conceptual, esta plataforma agrupa cada módulo en una carpeta correspondiente, dentro de moodle podemos crear cursos para ese apartado se utiliza la carpeta de “**course**” en ella se guarda tanto la creación de los cursos, la edición de ellos, como la vista de cada uno de los cursos.

Por otra parte en el caso del módulo de actividades la carpeta se llama “**Mod**” en ella están guardadas todas las actividades que se pueden realizar.

Dentro de esas actividades está la de las tareas, la cual la encontraremos con el nombre de “*assignment*” esta carpeta de tarea, dentro de ella cuenta con unos archivos PHP. Que son las configuraciones u opciones que se pueden hacer con ellas, los cuales modificamos para poder mandar los avisos a los usuarios.

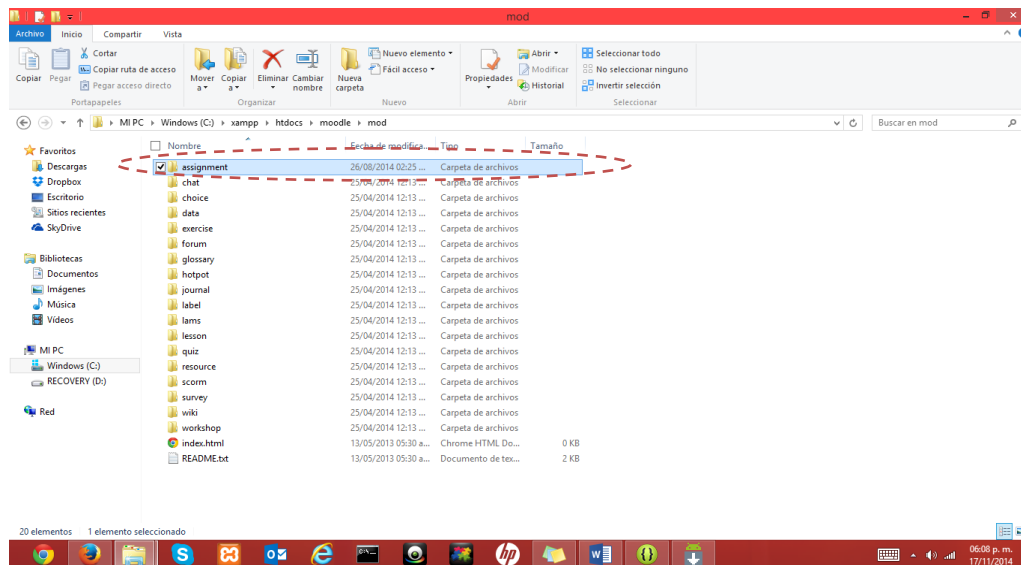


Figura 34. Muestra la carpeta de tarea dentro de moodle (*assignment*)

4.9.1 Consulta de base datos (probándolo en la vista del curso)

Como vimos anteriormente dentro de la carpeta *assignment* se encuentran archivos que son de configuraciones para el módulo de tareas de moodle, una de las primeras cosas que realizamos para poder hacer el envío de las notificaciones a los alumnos fue filtrar u obtener a los alumnos que solamente estuvieran en ese curso. Ya que recordemos que dentro de moodle están todos guardados en una sola tabla, ósea que en ella están todos los alumnos de los demás cursos.

Para ello se filtró a través del **ID_curso** de la materia (curso), una vez obteniendo a todos los usuarios que su **ID_curso** coincida con la de la materia entonces a ellos son los que solamente se mandara el aviso, además en esa consulta se obtiene el **ID_regid** que es el registro que el servidor GCM le dio al dispositivo Android. Esta consulta SQL se realizó en la vista del curso (VIEW.php). Dentro de la vista del curso de obtiene a los usuarios. Dependiente en que curso se encuentre el profesor.

Wiew.php

```
<?php // $Id$

// Display the course home page.

require_once('../config.php');
require_once('lib.php');
require_once($CFG->libdir.'/blocklib.php');
require_once($CFG->libdir.'/ajax/ajaxlib.php');
require_once($CFG->dirroot.'/mod/forum/lib.php');

$id      = optional_param('id', 0, PARAM_INT);
$name    = optional_param('name', '', PARAM_RAW);
$edit    = optional_param('edit', -1, PARAM_BOOL);
$hide    = optional_param('hide', 0, PARAM_INT);
$show    = optional_param('show', 0, PARAM_INT);
$idnumber = optional_param('idnumber', '', PARAM_RAW);
$section = optional_param('section', 0, PARAM_INT);
$move    = optional_param('move', 0, PARAM_INT);
$marker  = optional_param('marker', -1, PARAM_INT);
$switchrole = optional_param('switchrole', -1, PARAM_INT);

$sobtener_datos=mysql_query("SELECT DISTINCT gcm_regid FROM
mdl_notificaciones_movil WHERE email IN (SELECT email FROM
mdl_notificaciones_movil
WHERE ID_curso=(SELECT idnumber FROM mdl_course WHERE id=$id))");
echo $id.'
```

Figura 35. Muestra código que obtiene los datos de los alumnos de un curso específico

La consulta consiste en que se obtendrá a los alumnos de la tabla “*mdl_course*”, (que es allí donde están todos los alumnos de los cursos), esto es a través del id del curso una vez que aparezcan todos los alumnos que están en ese curso se comparara con la tabla “*mdl_notificaciones_movil*”, a través del email, esa tabla guarda los datos cuando los alumnos se registran a través de la aplicación y después solo mostraran a los alumnos con su registro de gcm_regid (servidor gcm). Que quiere decir que ya instalaron en sus dispositivos la aplicación. Y que solamente a ellos les llegaran los avisos.

Segundo paso:

Una vez que se obtuvo a los alumnos a quienes solamente les llegaran los avisos, se procedió a modificar la parte que realizara el envío del aviso a los dispositivos de esas personas, el aviso se hará cuando el profesor haya terminado de rellenar los datos del formulario de la tarea y cuando le dé clic a cualquiera de los dos botones para guardar (guardar y mostrar/guardar regresar al curso), el sistema guardara pero también realizara el envío a los estudiantes.

En esta parte nosotros tuvimos que agregar un código dentro del archivo de “*moodleform_mod.php*”, el cual se encuentra dentro de la carpeta de “course”, este archivo contiene la forma, los campos y eventos de los elementos del formulario que rellena el profesor cuando sube una tarea.

Nosotros agregamos el código que hace que se envíen las notificaciones a los alumnos, para ellos agregamos en la parte del evento de los botones de guardar (guardar cambios regresar al curso /guardar y mostrar), la opción que realiza el envío.

```

function add_action_buttons($cancel=true, $submitlabel=null, $submit2label=null)
{
    if (is_null($submitlabel)) {
        $submitlabel = get_string('savechangesanddisplay');
    }

    $obtener_datos=mysql_query("SELECT DISTINCT gcm_regid FROM
mdl_notificaciones_movil WHERE email IN (SELECT email FROM
mdl_notificaciones_movil
WHERE ID_curso=(SELECT idnumber FROM mdl_course WHERE id=$id))");

    //echo $id.'<br>';

    while($registro=mysql_fetch_array($obtener_datos)) {
        //echo $regId=$registro[0].'<br>';
        $regId=$registro[0];
    }

    include_once './gcm_server_php/GCM.php';
    $gcm = new GCM();
    $registatoin_ids = array($regId);
    $message = $name;
    $message = array("price" => $message);

    $result = $gcm->send_notification($registatoin_ids, $message);
    // echo $result;
}

```

Figura 36. Código del evento del primer botón de guardar, hace el envío del mensaje

En este primer evento del primer botón, una vez obtenido los alumnos que solamente estén ese grupo se procederá a realizar el envío el mensaje en donde hacemos uso del php GCM.

Mandando solamente lo que tiene el campo del formulario de la tarea el cual tiene como nombre (name), ese será enviado a los alumnos correspondientes, guardara la tarea y regresara al curso.

En el segundo botón de guardar y mostrar la tarea, se agregó al segundo evento el mismo código del botón anterior. Mandando solamente lo que tiene el campo del formulario de la tarea el cual tiene como nombre (name), ese será enviado a los alumnos correspondientes, guardara la tarea la mostrara en pantalla.

```
function add_action_buttons($cancel=true, $submitlabel=null, $submit2label=null)
{
    if (is_null($submitlabel)) {
        $submitlabel = get_string('savechangesanddisplay');
    }

    if (is_null($submit2label)) {
        $submit2label = get_string('savechangesandreturntocourse');
    }

    $obtener_datos=mysql_query("SELECT DISTINCT gcm_regid FROM
mdl_notificaciones_movil WHERE email IN (SELECT email FROM
mdl_notificaciones_movil
        WHERE ID_curso=(SELECT idnumber FROM mdl_course WHERE id=$id)");
    //echo $id.'<br>';

    while($registro=mysql_fetch_array($obtener_datos)) {
        //echo $regId=$registro[0].<br>';
        $regId=$registro[0];
    }

    include_once './gcm_server_php/GCM.php';

    $gcm = new GCM();

    $registatoin_ids = array($regId);

    $message = $name;

    $message = array("price" => $message);

    $result = $gcm->send_notification($registatoin_ids, $message);

    // echo $result;
```

Figura 37. Código del evento del segundo botón de guardar, hace el envío del mensaje

4.10. Resultado de las pruebas

En este apartado visualizaremos el resultado que obtuvimos de nuestra aplicación, tanto la aplicación desarrollada en *Android* como la de *moodle*.

4.10.1 Interfaz del registro de usuario

La interfaz de registro en el dispositivo con *Android*, muestra una pantalla con un pequeño formulario, en donde se piden el nombre de la persona, seguidamente de su correo electrónico, cabe mencionar que este correo debe ser de Gmail, posteriormente pide el ID_curso que este será el identificador de la materia en la que el alumno está inscrito, este identificador será proporcionado al alumno ya sea por parte del administrador de la plataforma o del profesor del curso.

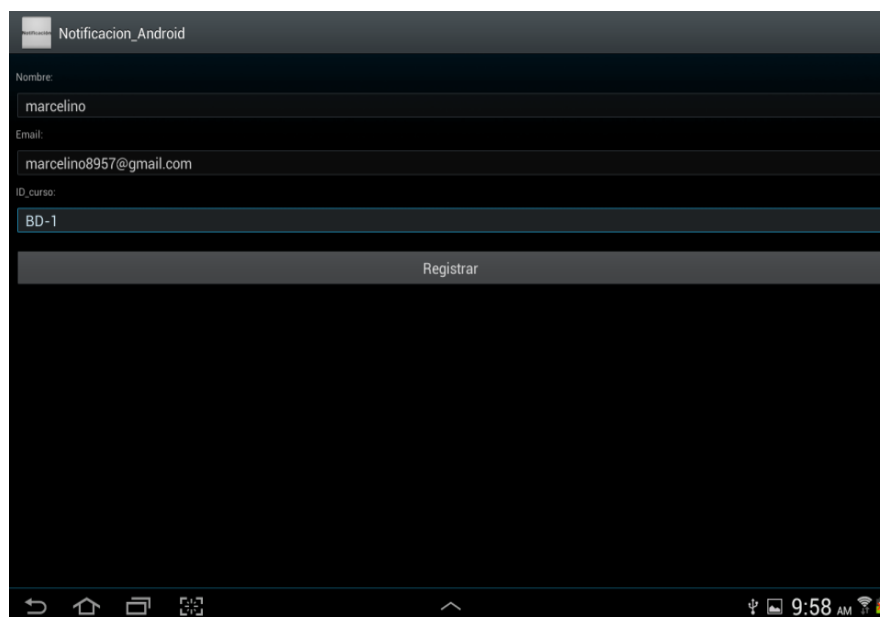


Figura 38. Pantalla principal de la aplicación Android.

4.10.2 Interfaz de conexión al servidor, en espera de mensajes

Una vez que el usuario haya dado clic en el botón de registrar con los datos debidamente proporcionados, esos datos se almacenan en la base de datos que se encuentra en *moodle*

para su registro y la aplicación registrará al usuario y realizará la conexión al servidor de GCM, para poder hacer las notificaciones y en espera de que estas lleguen al dispositivo.



Figura 39. Muestra la conexión al servidor de GCM

4.10.3 Interfaz del dispositivo, recibiendo una notificación

Esta pantalla muestra cuando el usuario está en espera de recibir alguna notificación (mensaje), cuando desde la página de *moodle* se sube la tarea y el dispositivo lo recibe de la manera que se muestra a continuación.

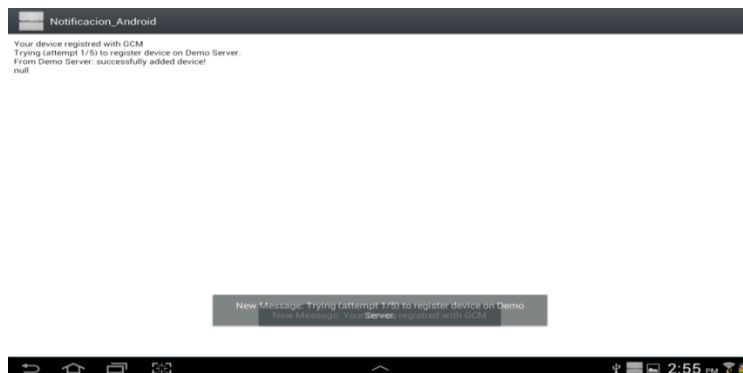


Figura 40. Muestra la llegada de una nueva notificación.

4.10.4 Interfaz del dispositivo, llegada de la notificación

Es esta pantalla se muestra la llegada del mensaje al dispositivo móvil desde la página de moodle notificándole que hay una tarea para realizar y el nombre de la tarea, subida.

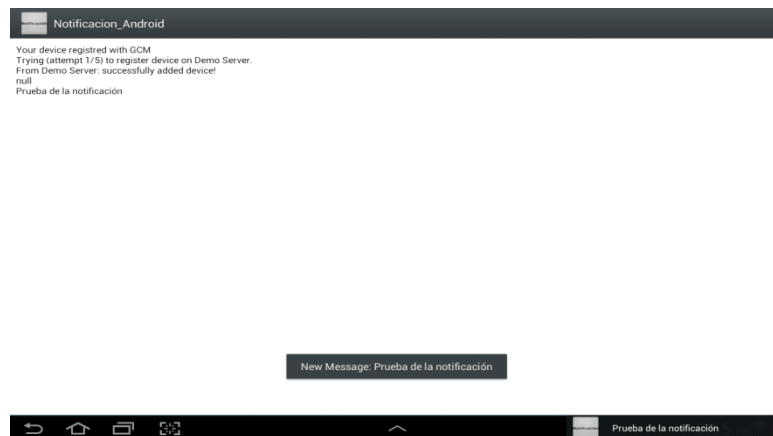


Figura 41. Muestra el mensaje en el dispositivo

4.10.5 Interfaz de moodle, donde se elige el tipo de tarea

Esta pantalla muestra los 3 tipos de tareas de la plataforma de moodle, las cuales mandarían la notificación al usuario.

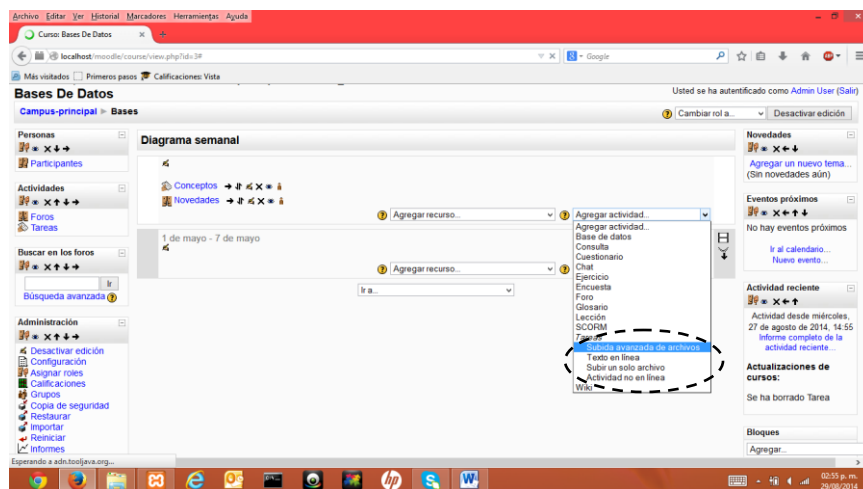


Figura 42. Muestra los tipos de tareas en moodle

4.10.6 Interfaz de moodle, donde se realiza la carga y el envío del mensaje

Una vez elegido el tipo de tarea, se muestra la siguiente pantalla en donde se coloca el nombre de la tarea, la descripción y algunos otros datos acerca de dicha actividad, una vez que se haya llenado el formulario, existen dos botones para guardar (Guardar cambios y regresar al curso o Guardar cambios y mostrar) y el de cancelar.

Cualquiera de los dos botones de guardar, hace el envío del mensaje al dispositivo, cabe decir que solamente se enviará el campo del formulario que dice nombre de la tarea.

Archivo Editar Ver Historial Marcadores Herramientas Ayuda

Bases: Editando Tarea

localhost/moodle/course/modedit.php?add=assignment&type=uploadsingle&course=3§ion=0&return=0

Más visitados Primeros pasos Calificaciones: Vista

Bases De Datos Usted se ha autenticado como Admin User (Salir)

Campus-principal > Bases > Tareas > Editando Tarea

Agregando Tarea

Ajustes generales

Nombre de la tarea* Prueba de la notificación

Descripción

Trebuchet 1 (8 pt) Idioma B I U S x2 x1 < > < >

Esta es una prueba para comprobar el envío del aviso acerca de la tarea, a los alumnos.

Ruta: body

Calificación 100

Disponibilidad 15 agosto 2014 14 55 Deshabilitar

Fecha de entrega 5 septiembre 2014 14 55 Deshabilitar

Impedir envíos retrasados No

Subir un solo archivo

Permitir reenvío No

Alertas de email a los profesores No

Tamaño máximo 2Mb

Ajustes comunes del módulo

Modo de grupo No hay grupos

Visible Mostrar

Número ID

Categoría de calificación actual: Sin categorizar

Guardar cambios y regresar al curso Guardar cambios y mostrar Cancelar

En este formulario hay campos obligatorios

Moodle Docs para esta página

Usted se ha autenticado como Admin User (Salir)

Bases

Figura 43. Muestra el formulario de la tarea a enviar

4.10.7 Interfaz de moodle, tarea subida

Aquí se muestra cuando la tarea ya ha sido subida y por consiguiente la alerta de la notificación ya fue enviada.

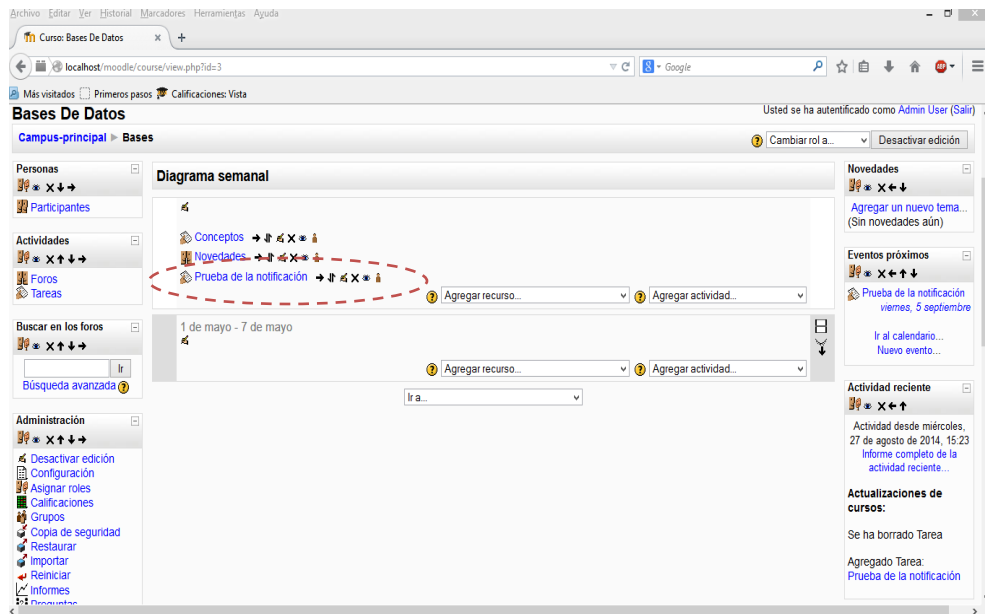


Figura 44. Muestra la pantalla con la tarea subida

CAPÍTULO 5
CONCLUSIÓN. TRABAJO FUTURO

5.1 Conclusión

Sin duda alguna hoy en día la tecnología ha hecho que las cosas sean mucho más fáciles en cualquiera que sea el entorno, educativo, comercial, empresarial, etc. Específicamente en la educación ha contribuido en la forma de enseñar e impartir clases, ya que gracias a ella, se cuenta hoy en día con una modalidad de enseñanza en la que el alumno no necesita asistir físicamente a un aula de clases. Al utilizar alguna aula virtual de aprendizaje (AVA), en ella se puede subir tareas, actividades, etc. Y poder así evaluar o retroalimentar al alumno, la dificultad o problemática es que el alumno solamente conoce y sabe si existe alguna tarea o actividad siempre y cuando él, entre al aula virtual a través de internet. Con el trabajo de tesis que se desarrolló, se divulga un dicho tecnológico para esta modalidad de enseñanza llamada *online*, ya que fortalece la comunicación de las actividades marcadas por el profesor a los alumnos, haciendo que a través del celular o dispositivo (*Android*), se le notifique al alumno que se ha subido una tarea para realizar, ya que realmente en la práctica el estudiante se entera de las actividades o tareas, gracias a que la aplicación desarrollada que permite guardar la tarea y mandar el título de la tarea a través de un pequeño mensaje al dispositivo. Las notificaciones son recibidas por aquellas personas que están registradas en el curso o materia.

Además de que este trabajo de tesis puede ser una muy buena aportación para la comunidad de *moodle*, toda vez que esta plataforma o aula virtual es de código abierto, lo cual permite que se modifique su código y se anexas otros componentes, para el beneficio de la comunidad.

Por último quiero mencionar que hoy en día la educación ha cambiado la forma tradicional de enseñar, y con ello poder brindarles a otras personas una alternativa más para estudiar y poder mejorar la calidad de vida y el nivel educativo en nuestro país.

5.2 Trabajo a futuro

Como trabajo o meta a futuro estamos considerando algunos puntos que pensamos que serían importantes que se anexaran al proyecto para hacer un poco más completa la aplicación, algunos de estos puntos son por ejemplo:

- Que se guarden las notificaciones o mensajes en alguna tabla en el dispositivo, para que el alumno pueda tener un historial de los mensajes recibidos.
- Por otra parte se quiere hacer esta aplicación tipo *plugin*, el cual consiste en que a la hora de instalarlo automáticamente se configuran los archivos como (clases, tablas, etc.) que se utilizan para generar o habilitar las notificaciones.
- También se quiere desarrollar esta aplicación para otros sistemas operativos.

REFERENCIAS

- Arjona Heredia, J. E., & Gámiz Sanchez, V. M. (2013). *Revisión de opciones para el uso de la plataforma moodle en dispositivos móviles*. Recuperado el 30 de Septiembre de 2013, de <http://www.redalyc.org/articulo.oa?id=54726040005>
- Baño Sancho, J. (2007). *La plataforma Educativa moodle*. Recuperado el 24 de Marzo de 2014, de http://www.fvet.uba.ar/postgrado/Moodle18_Manual_Prof_1.pdf
- Barberá, E., & Badia, A. (22 de Noviembre de 2005). *El uso educativo de las aulas virtuales emergentes en la educación superior*. Recuperado el 22 de Septiembre de 2013, de <http://www.uoc.edu/rusc/2/2/dt/esp/barbera.pdf>
- Belloch, C. (2010). *Las actividades en los EVA*. Recuperado el 26 de Marzo de 2014, de <http://www.uv.es/bellohc/pedagogia/EVA7.pdf>
- Boneu, J. M. (Abril de 2007). *Plataformas abiertas de e-learning para el soporte de contenidos educativos abiertos*. Recuperado el 23 de septiembre de 2013, de <http://rusc.uoc.edu/index.php/rusc/article/viewFile/v4n1-boneu/v4n1-boneu>
- Davila, A. (Julio de 2011). *Filosofía educativa de las aulas virtuales*. Recuperado el 29 de Septiembre de 2013, de <http://www.redalyc.org/articulo.oa?id=88024213006>
- Fernandez Pampillón, A. C. (2007). *La plataforma e-learning para la enseñanza y el aprendizaje universitario en internet*. Recuperado el 12 de septiembre de 2013, de http://eprints.ucm.es/10682/1/capituloE_learning.pdf
- Garcia Sanchez, J., & Castillo Rosas, A. (Septiembre de 2005). *Los componentes de un sistema de educación virtual: El subsistema tecnologico*. Recuperado el 28 de Septiembre de 2013, de <http://www.odiseo.com.mx/2005/07/garcia-castillo-componentetecnologico.htm>
- Llorente Cejudo, M. C. (Marzo de 2007). *Moodle como entorno virtual de formación al alcance de todos*. Recuperado el 28 de Septiembre de 2013, de <http://www.redalyc.org/articulo.oa?id=15802827#>
- López Ramos, E. R. (s.f.). *Utilizaciones de la plataforma Interactiva (moodle) como medio de enseñanza y comunicación*. Recuperado el 29 de Septiembre de 2013, de <http://www.ilustrados.com/tema/12755/Utilizacion-Plataforma-Interactiva-Moodle-como-medio.html>

- Malave Polanco, K., & Beauperthuy Taibo, J. L. (Julio de 2011). *"Android" el sistema operativo de google para dispositivos móviles*. Recuperado el 13 de Septiembre de 2013, de <http://www.redalyc.org/articulo.oa?id=78219156004#>
- Martinez De Lahidalga, I. R. (2008). *Moodle, la plataforma para la enseñanza y organización escolar*. Recuperado el 29 de Septiembre de 2013, de http://www.ehu.es/ikastorratza/2_alea/moodle.pdf
- Miranda Díaz, G. A. (10 de noviembre de 2004). *De los ambientes virtuales de aprendizaje a las comunidades de aprendizaje en línea*. Recuperado el 12 de septiembre de 2013, de http://www.revista.unam.mx/vol.5/num10/art62/nov_art62.pdf>, Consultado: 12 septiembre-2013
- Ospina P, D. (2008). *Introducción a moodle*. Recuperado el 25 de Marzo de 2014, de <http://aprendeenlinea.udea.edu.co/boa/contenidos.php/ef7ad51ac40fc83cbaf9e4d9d2373f9b/175/estilo/aHR0cDovL2FwcmVuZGVlbnxpbmVhLnVkZWEuZWR1LmNvL2VzdGlsb3MvYXp1bF9jb3Jwb3JhdGl2by5jc3M=/1/contenido/imagenes/Moodle.jpg>
- Sanchez Rodriguez, J. (Enero de 2009). *plataformas virtuales de aprendizaje*. Recuperado el 12 de 09 de 2013, de http://www.revistavirtualpro.com/ediciones/gestion_de_conocimiento_en_ambientes_virtuales_de_aprendizaje_ava_aplicado_a_procesos_industriales_plataformas_virtuales_de_aprendizaje-2011-11-01_8
- Vicent, A. (2004). *Mapas Conceptuales de moodle*. Valencia España.

ÍNDICE DE FIGURAS

Figura 1. Cronograma de actividades.	21
Figura 2. El libro modelo de agregación de contenidos SCORM como parte de la librería	27
Figura 3. Pantalla de la aplicación	46
Figura 4. Pantalla de la aplicación	47
Figura 5. Visión General de los recursos de Moodle (Ospina P, 2008).	50
Figura 6. Muestra el módulo de las actividades de Moodle (Vicent, 2004).	51
Figura 7. Muestra los recursos del módulo de actividades tareas (Vicent, 2004).	52
Figura 8. Muestra el Diagrama conceptual del módulo de tareas.....	54
Figura 9. Muestra los datos de cada tarea.	55
Figura 10. Muestra el diagrama de clases y el atributo de las tareas.....	58
Figura 11. Casos de uso relacionado con la tarea	59
Figura 12. Diagrama de robustez de crear una tarea	61
Figura 13. Diagrama de robustez de editar una tarea	62
Figura 14. Diagrama de robustez de eliminar una tarea	63
Figura 15. Diagrama de secuencia para crear una tarea	64
Figura 16. Diagrama de secuencia para editar una tarea	65
Figura 17. Diagrama de secuencia para eliminar una tarea	66
Figura 18. Muestra la aplicación en eclipse y sus clases	68
Figura 19. Muestra el código de la clase “AlertDialogManager.java”	68
Figura 20. Muestra el código de la clase “CommonUtilities.java”	69
Figura 21. Muestra el código de la clase “ConnectionDetector.java”	69
Figura 22. Muestra el código de la clase “GCMIntentService.java”	70
Figura 23. Muestra el código de “ServerUtilities.java”	71
Figura 24. Pantalla principal de la aplicación en Android.....	72
Figura 25. Código que muestra la primera pantalla de la aplicación Android.	72
Figura 26. segunda pantalla de la aplicación en Android	73
Figura 27. Muestra el Código de la pantalla de conexión al servidor	73
Figura 28. Muestra la sentencia para crear la tabla en la base de datos moodle	74
Figura 29. Muestra la configuración hacia la base datos moodle.....	74
Figura 30. Código para las conexiones de base de datos.....	75
Figura 31. Código para la creación de usuario único	76
Figura 32. Código para enviar solicitudes de notificación de inserción a servidor GCM...77	
Figura 33. Código para almacenar el usuario a la base datos	78
Figura 34. Muestra la carpeta de tarea dentro de moodle (assignment)	79
Figura 35. Muestra código que obtiene los datos de los alumnos de un curso específico....80	
Figura 36. Código del evento del primer botón de guardar, hace el envío del mensaje.....82	

Figura 37. Código del evento del segundo botón de guardar, hace el envío del mensaje	83
Figura 38. Pantalla principal de la aplicación Android.	84
Figura 39. Muestra la conexión al servidor de GCM	85
Figura 40. Muestra la llegada de una nueva notificación.	85
Figura 41. Muestra el mensaje en el dispositivo.....	86
Figura 42. Muestra los tipos de tareas en moodle	86
Figura 43. Muestra el formulario de la tarea a enviar.....	87
Figura 44. Muestra la pantalla con la tarea subida	88

ÍNDICE DE TABLAS

Tabla 1. Comparativa entre sistemas CBT y sistemas e-learning.....	25
Tabla 2. Resumen comparativo de los LMS y LCMS	29
Tabla 3. Muestra las actividades que existen en moodle	49
Tabla 4. Muestra cuando se crea una tarea	59
Tabla 5. Muestra cuando se edita una tarea	60
Tabla 6. Muestra cuando se elimina una tarea	60
Tabla 7. Muestra cuando se consulta una tarea.....	60
Tabla 8. Muestra las características cuando se sube una tarea	61
Tabla 9. Muestra las características cuando se edita una tarea	62
Tabla 10. Muestra las características cuando se elimina una tarea.....	63
Tabla 11. Muestra las clases del proyecto de Android.	67