

Utilizando herramientas de software libre para la gestión de cursos de grado

Francisco J. Díaz, Claudia M. Banchoff Tzancoff, Ana P. Amadeo, Einar F. Lanfranco

Laboratorio de Investigación en Nuevas Tecnologías Informáticas – LINTI, Facultad de Informática – Universidad Nacional de La Plata {javierd, cbanchoff, pamadeo, einar}@info.unlp.edu.ar

Resumen

La cátedra "Proyecto de Software" es una materia donde los alumnos desarrollan aplicaciones web usando PHP. Cuenta con un promedio de 180 alumnos por año y, por la etapa de la carrera donde se encuentra, muchos de ellos trabajan. Esta cátedra abarca una gran cantidad de temas que se abordan en forma práctica desarrollando trabajos.

Dadas estas particularidades, desde hace cuatro años se utilizan herramientas de elearning como apoyo a las clases presenciales, y desde el año pasado herramientas para el manejo de versiones a fin de organizar las entregas de los trabajos prácticos y agilizar el proceso de corrección.

El objetivo del presente artículo es presentar la experiencia en el uso de algunas herramientas de código abierto durante el dictado de la materia Proyecto de Software. Se presentarán tanto las herramientas de gestión utilizadas como las que se proponen dentro de los contenidos curriculares. Se analizará la metodología utilizada y los resultados obtenidos en la cursada 2008.

Palabras clave: Software Libre, Moodle, SVN, curso de grado, aplicaciones web

1. Introducción

La materia "Proyecto de Software" se dicta en el 3º año de la carrera Analista Programador Universitario y las Licenciaturas de Informática y de Sistemas de la Facultad de Informática de la Universidad Nacional de la Plata.

Es un materia cuatrimestral, cuyo objetivo principal es introducir al alumno los conceptos básicos para poder

desarrollar aplicaciones web usando PHP[1]. Los temas abordados comienzan con los estándares HTML/XHTML[2], CSS[3], XML[4], XSL[5] y se trabaja tanto en la programación en el cliente como en el servidor. En el lado del cliente, se programa utilizando el lenguaje Javascript [6] y en el servidor se trabaja con PHP.

Las aplicaciones desarrolladas generalmente acceden a bases de datos y para ello se utiliza MySQL[7] y otras herramientas que se describirán en la siguiente sección. Proyecto de Software tiene como correlativas, entre otras: Ingeniería de Software e Introducción a las Bases de Datos, con lo que el alumno ya tiene los conceptos teóricos necesarios para ponerlos en práctica en el desarrollo de una aplicación específica. Se introduce también el tema de seguridad en aplicaciones Web, dando unas mínimas guías y consejos para el desarrollador PHP.

2. Metodología y herramientas utilizadas

El dictado de la materia se divide en clases teóricas y prácticas. Las clases teóricas son de carácter expositivo, utilizando presentaciones para reforzar los conceptos básicos y ejemplos concretos mediante el uso de una PC y un proyector. Se hace especial hincapié en mostrar las distintas implementaciones "en vivo", dado que parte de los contenidos se centra también en la instalación y configuración de algunos servicios de base como ser un servidor Apache.

Las clases prácticas se desarrollan en las salas de PCs de la Facultad. Cabe aclarar que la Facultad cuenta con 3 salas de cómputos habilitadas a las cátedras de grado, equipadas con aproximadamente de 15 a 20 máquinas cada una. Dos de ellas cuentan con máquinas con doble booteo, pudiéndose utilizar tanto Microsoft Windows como Lihuen GNU/LINUX[8]. Desde la cátedra no se

hace ningún requerimiento específico para que trabajen sobre uno u otro sistema operativo. Se les exige que sus desarrollos puedan utilizarse tanto desde un entorno Windows como desde uno Linux, y, dado que se trata de aplicaciones ejecutadas desde un navegador web, es indispensable también que puedan ser visualizadas en los navegadores más populares.

Los alumnos concurren a las consultas prácticas ya sea para desarrollar "in situ" sus aplicaciones o utilizar las computadoras y la conexión a Internet para acceder y "subir" sus tareas a nuestras plataformas de trabajo.

Además de un examen parcial, la cursada se aprueba con la entrega de un trabajo realizado en cinco (5) etapas. De estas cinco entregas, se deben aprobar 3 (tres) incluyendo la última, en la que se aplican todos los conceptos aprendidos.

Cada entrega agrega distintas herramientas y dificultades, pero todas parten de un mismo tema.

La primer entrega corresponde a un prototipo básico de la aplicación y la última es la aplicación terminada. Cada entrega se defiende con un coloquio presencial obligatorio, lo que implica un contacto directo entre docentes y alumnos.

Para la entrega de estos trabajos, los alumnos conforman grupos de hasta dos personas, conformando un promedio de 100 grupos por año. En el caso de la cursada 2008, se trabajó con 98 grupos, con un total de 192 inscriptos.

El plantel docente de Proyecto de Software está compuesto por dos profesoras, dos jefes de trabajos prácticos y 10 ayudantes. Cada ayudante tiene a su cargo la corrección y seguimiento de cierta cantidad de grupos y todos son supervisados por los dos jefes de trabajos prácticos. En el caso de los ayudantes más novatos en el cargo, corrigen junto con otros de mayor experiencia, por lo que la cantidad de grupos asignados a cada ayudante varía entre 5 y 20 trabajos cada uno.

Para evitar confusiones en las correcciones y homogeneizar las versiones a utilizar, la cátedra montó un servidor web con todas las herramientas que se utilizaron en el desarrollo de las aplicaciones. En la cursada 2008, se instaló un servidor Apache, PHP v5.2, MySQL v los paquetes PEAR::DB[9], y el sistema de templates Smarty[10].

En esta cátedra se trabaja con PHP desde el año 1999, siendo unos de los pioneros en utilizar este lenguaje[11] y, desde entonces se hace especial hincapié en la utilización de herramientas de código abierto, tanto para la implementación, como en las de soporte. En otros años, no sólo se utilizaron herramientas de código

abierto, sino que algunas de las aplicaciones realizadas por los alumnos fueron liberadas a la comunidad de software libre. El caso más emblemático es el de SAE_Admin[12], desarrollada en la cursada 2004 y mejorada en la cursada 2006. En la cursada 2008, se implementó un sistema de creación y gestión de encuestas en la web y aún no hemos evaluado la posibilidad de liberar algunas de las aplicaciones.

Tanto la gran cantidad de alumnos y grupos como la metodología adoptada de 5 entregas en una materia cuatrimestral, hacen que sea indispensable la utilización de herramientas que asistan a la gestión y seguimiento de los alumnos. Es así que desde hace cuatro años se utiliza Moodle como plataforma de apoyo a la cursada presencial y el año pasado se adopta un servidor de manejo de versiones para la gestión y seguimiento de las implementaciones y entregas de los alumnos.

La materia se aprueba con la entrega de un trabajo adicional, si es que el alumno quiere adherir al régimen de promoción o, de forma tradicional, rindiendo un examen final. Es interesante recalcar que si bien es más cómodo, tanto para los docentes como para los alumnos, rendir un examen final, el 96% elige realizar este trabajo, que demanda más tiempo pero que permite comprobar y afianzar mucho más todos los conceptos aprendidos.

Este año, los alumnos pueden optar entre opciones distintas para su trabajo de promoción. El primero de ellos cubre el análisis de librerías Javascript para la programación client side como Prototype, Dojo, etc. Otro de los trabajos prevé la utilización de un framework MVC[30] para PHP. Los alumnos deben realizar una pequeña implementación usando CakePHP o Symfony. Por último, otra opción planteada fue la de un análisis de vulnerabilidades de seguridad sobre sus propios trabajos realizados durante la cursada.

A continuación detallaremos las distintas herramientas y la forma en que las mismas fueron utilizadas en nuestra cátedra.

Moodle



Moodle[15] es un VLE[16] (Virtual Learning Environment) de código abierto, desarrollado a partir del año 2001 por Martin Dougiamas, como tesis doctoral en Curtin University of Technology, Perth, Australia, y finalizada en el 2003. La propuesta surge a partir de la necesidad de disponer de un sistema open source para la gestión de cursos a través de la Web. A partir de ese momento el desarrollo de Moodle fue exponencial. Su comunidad de usuarios, actualmente cuenta con más de 677.632 usuarios registrados, y el dinamismo de su grupo de desarrolladores[18], hacen que Moodle sea considerado el Linux de los sistemas de gestión de contenidos[17].

Actualmente se encuentra traducido a 78 idiomas en 209 países, la última versión estable es la 1.9.4 liberada el 24 de enero del 2009. Este sistema incorpora, además de las características básicas deseables en este tipo de entornos, como publicación de contenidos en diferentes formatos, autoevaluaciones, entrega de trabajos prácticos en línea, con fecha límite y correcciones en línea, cuestionarios, consultas, herramienta para la construcción de evaluaciones HotPotatos, reportes, entre otras; la posibilidad de definir distintos modelos pedagógicos, soporte de estándares y metadatos (soporta SCORM 1.2[13] y LAMS 1.4[14]), y un conjunto de aplicaciones sociales básicas, como wikis, blogs, canales RSS, entre otros.

En la Facultad de Informática se comenzó a utilizar a partir del año 2003, como soporte de cursos presenciales y semi-presenciales. Actualmente cuenta con más de 2200 usuarios, más de 100 cursos y sólo en el mes de agosto, los accesos ascendieron a más de 4000. Como se mencionó previamente, la cátedra de Proyecto de Software comenzó a utilizar la plataforma en modalidad blended learning o como complemento de las clases presenciales, desde el año 2005. En este período el entorno se utilizó principalmente para distribuir el material teórico y el enunciado de los trabajos prácticos.

Actualmente se utiliza, además de repositorio de los materiales digitales, para realizar distintas actividades y la coordinación de las mismas. En la Figura 1 se puede observar la estructura general del curso en la plataforma, las actividades propuestas, el contenido digital y elementos de coordinación.

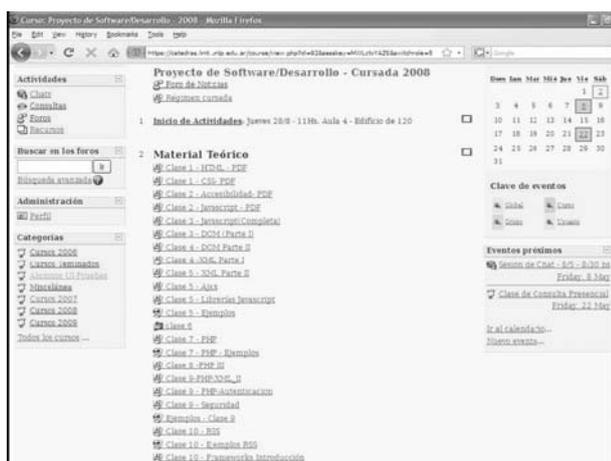


Figura 1 – Página inicial del curso en vista de un alumno, donde se destaca la sección de Actividades, Contenido y Coordinación.

Respecto a las **actividades**, durante el año 2008 se utilizaron los foros de discusión de una forma mucho más intensiva que cursadas anteriores. Se postearon más

de 800 mensajes y se registran más de 40000 consultas o vistas de los mismos. Las consultas en línea permitieron resolver cuestiones administrativas de una forma rápida y eficiente. Por ejemplo los estudiantes seleccionaron el turno de la práctica y confirmaban su asistencia a los parciales a través de esta herramienta (ver Figura 2).



Figura 2 – Actividades Consulta

En promedio, las consultas fueron accedidas, entre vistas y modificaciones, más de 700 veces.

Se utilizaron sesiones de chat para las coordinación de los trabajos de promoción. Como se mencionó previamente, los estudiantes que optan por realizar la promoción, para obtener la aprobación de la materia deben realizar un trabajo integrador y rendir un coloquio. A través de los encuentros presenciales y virtuales los estudiantes continúan en contacto con la cátedra, llegando a la instancia de la entrega final en forma progresiva.

Respecto al contenido digital, como mencionamos previamente, en la plataforma los docentes ponen a disposición el material de las clases teóricas (presentaciones y ejemplos) en diferentes formatos, las explicaciones de práctica y material adicional que consideran relevante, como URLs de interés y tutoriales en línea.

En cuento a la coordinación, el calendario de la plataforma se utilizó para informar a los alumnos sobre las fechas de parciales, entregas y encuentros virtuales. A través de los roles, estudiante, profesor, jefe de trabajos prácticos y ayudantes, cada participante accede a funciones específicas. Los informes de actividad y accesos, a nivel del curso y de cada recurso en particular, permitieron medir el grado de apropiación de cada uno de ellos (ver Figura 3).



Figura 3 – Informes

El recurso Tarea para centralizar las entregas y correcciones no se implementó en ninguna de las promociones. Pero a partir de este año se utilizó el sistema para el manejo de versiones Subversión[19] para agilizar las entregas y las correcciones de las mismas.

Subversion



El sistema Subversion, también conocido como SVN, es un software de sistema de control de versiones de cualquier documento digital. Mantiene el registro de todo el trabajo y los cambios en los archivos (código fuente principalmente) que forman un proyecto. Agiliza, ordena y promueve la colaboración entre distintos miembros del proyecto. El grupo de desarrolladores comenzó a trabajar en junio del 2000 y en mayo de 2002 se libera la primera versión, la 0.12.0. La última versión estable es la 1.6.1, liberada en abril de este año [20]. Es utilizado como herramienta de soporte de productos open source como ASF: The Apache Software Foundation [21], GNOME [22], PuTTY [23], Zope [24] entre otros.

Implementa una arquitectura cliente-servidor, que le permite organizar un repositorio central accedido y actualizado a través de clientes específicos.

Cada cliente mantiene una copia local y a través de comandos específicos la sincroniza con el repositorio tantas veces como sea necesario. Cada documento modificado en el directorio local se marca con un icono que nos indica el estado de la copia. Al registrar los cambios en el servidor se considera una revisión.

Actualmente existen numerosos clientes que implementan la interfaz a Subversion, como TortoiseSVN, que provee integración con el explorador de Windows. Es la interfaz más popular en este sistema operativo. [25], SVN: es el cliente de línea de comando de subversion directamente. Subclipse. [26], WorkBench[27] y KDESvn [28] entre otros. Particularmente en la cátedra se sugirió el uso de TortoiseSVN.

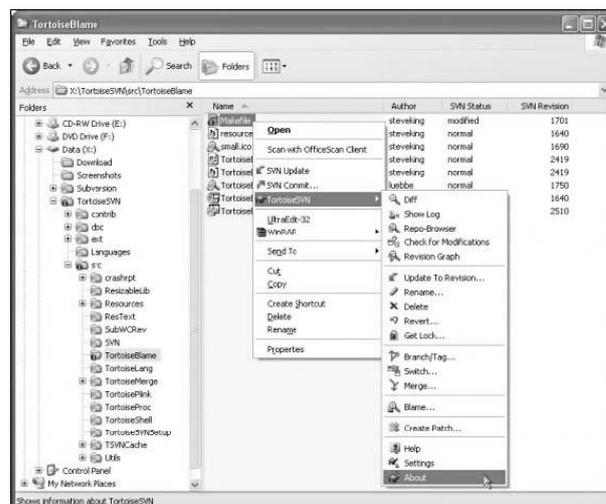


Figura 4 – Cliente TortoiseSVN

Como se mencionó previamente, para aprobar la cursada los estudiantes debieron realizar un trabajo final con 5 entregas parciales. Cada grupo fue definido como un usuario en el SVN, con acceso a un directorio y una base de datos específica. A través del cliente el estudiante sincronizaba sus archivos, subiendo y actualizando las versiones según su necesidad. Se utilizó phpMyAdmin[29] para que cada grupo accediera a su propia base de datos. A través de una dirección Web específica, que incluía su número de grupo, los alumnos accedían a la vista del usuario de su sistema.

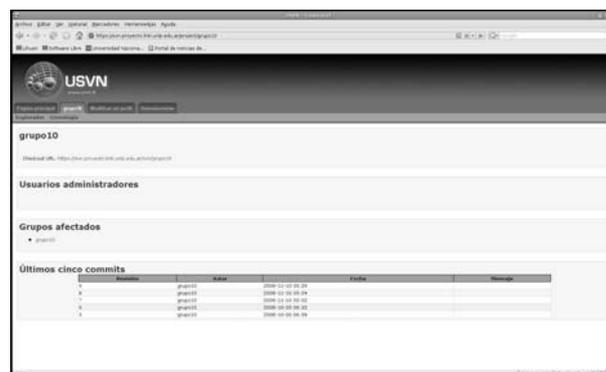


Figura 5 – Información sobre la actividad de un grupo a través del servidor SVN

Esta estructura también era utilizada por los ayudantes para realizar las correcciones. Cada ayudante tenía asignado un conjunto de grupos y tenía acceso al SVN de cada uno de ellos. En total se definieron 98 grupos, y se realizaron más de 400 entregas a través de esta herramienta.

Resultados obtenidos

De un total de 192 alumnos inscriptos, 133 participaron activamente y formaron 98 grupos para las entregas de los trabajos. Cabe aclarar que 3 de estos grupos nunca participaron, con lo que se trabajó realmente con 95 grupos. Los alumnos tuvieron 5 horarios de consulta en la semana y las entregas se completaron con un coloquio presencial en estos horarios. La Tabla 1 muestra la cantidad de alumnos presentados y aprobados en cada entrega.

	Desaprobados en el coloquio	Aprobados en la corrección general
1ra entrega	4	140
2da entrega	3	121
3ra entrega	2	105
4ta entrega	4	91
5ta entrega	3	43
Recuperatorio de la entrega final	0	75

Tabla 1 - Resultados de las entregas

Al finalizar la cursada, 118 alumnos lograron aprobar las entregas de los trabajos.

Lograr un 60% de aprobados se debe fundamentalmente a la guía constante por parte de los ayudantes.

Los alumnos no sólo debieron aprobar estos trabajos, sino que, además, rindieron un parcial tradicional. La Tabla 2 muestra los resultados obtenidos de los exámenes parciales.

	Presentados	Aprobados
1er parcial	127	63%
2do parcial	52	45.23
3er parcial	29	34.48
Total	109	66.87

Tabla 2 – Resultados de los parciales

Teniendo en cuenta las condiciones de aprobación de la cursada 101 alumnos aprobaron la materia, es decir un 53% de los inscriptos.

Considerando las experiencias de años anteriores, es un muy buen porcentaje de aprobación.

La materia se puede promocionar desarrollando un trabajo adicional. Pudiendo rendir un examen final y obtener así la materia, el 96% escogió realizar un trabajo especial, considerando que es la mejor forma de

consolidar y afianzar los conceptos aprendido.

El manejo de tantos alumnos y la coordinación de un plantel docente con diversas características hace que la utilización de herramientas que den soporte a estas actividades sea fundamental.

Si bien estos mismos conceptos se podrían impartir de otra manera, consideramos que la metodología adoptada es la más adecuada tanto desde el punto de vista de los alumnos, quienes tienen que respetar un cronograma estricto de entregas, sino también para los docentes, quienes van guiando y aconsejando a los grupos en el momento donde todavía está a tiempo de recuperarse, si es que no está llevando la materia como es debido.

Conclusión

El uso de sistemas de software libre en educación amplía en forma considerable la posibilidad de desarrollo de los estudiantes, y los motiva a participar en iniciativas similares. El uso de productos como SVN y Moodle facilitan el trabajo diario de correcciones y retroalimentaciones, particularmente en cursos como el caso presentado, por la cantidad de alumnos inscriptos y el número de trabajos a desarrollar, disminuyendo en forma considerable el tiempo de respuesta de las correcciones.

Tener una primera aproximación a productos como SVN durante la cursada le agrega un valor adicional a todo el ciclo de desarrollo de software en sí mismo. El uso de la plataforma virtual aumenta el grado de producción de los estudiantes, incluso en días feriados y recesos ya que los ayudantes contestaban las preguntas en forma no presencial. Si bien el porcentaje de aprobados aumentó ligeramente en este último año, la importancia del seguimiento y de contar con las herramientas necesarias hace que las tareas docentes sean más productivas..

Referencias

- [1]<http://www.php.net>
- [2]<http://www.w3.org/html/>
- [3]<http://www.w3.org/Style/CSS/>
- [4] <http://www.w3.org/XML/>
- [5] <http://www.w3.org/Style/XSL/>
- [6]<https://developer.mozilla.org/es/JavaScript>
- [7]<http://www.mysql.com/>

- [8]<http://lihuen.linti.unlp.edu.ar/>
- [9]<http://pear.php.net/>
- [10]<http://www.smarty.net/>
- [11]"PHP: una solución "open source" para el desarrollo de páginas Web dinámicas". Lic. Francisco Javier Díaz, Lic. Claudia Banchoff. Aceptado para su exposición en CACIC '2000.
- [12]<http://sourceforge.net/projects/sae-admin>
- [13]<http://es.wikipedia.org/wiki/SCORM>
- [14]<http://www.lamsinternational.com/>
- [15] <http://moodle.org>
- [16]<http://vlex.com/vid/aprendizajeplataformas-learning-contexto-200498>
- [17] <http://aarciathorgalson.wordpress.com/tag/moodle-18-web-20-elearning/>
- [18]<http://moodle.org/stats/>
- [19]<http://subversion.tigris.org/>
- [20]<http://www.tigris.org/servlets/ProjectNewsList>
- [21]<http://www.apache.org/>
- [22]<http://www.gnome.org/>
- [23]<http://www.chiark.greenend.org.uk/~sgtatham/putty>
- [24]<http://www.zope.org/>
- [25]<http://tortoissvn.tigris.org/>
- [26]<http://subclipse.tigris.org/>
- [27]<http://pysvn.tigris.org/>
- [28]<http://www.alwinworld.de/wiki/programs/kdesvn>
- [29]<http://www.phpmyadmin.net>
- [30]<http://www.ibm.com/developerworks/opensource/library/os-php-fwk1/>

Dirección de Contacto del Autor/es:

Francisco Javier Díaz
Calle 50 y 120. 2do. Piso
La Plata (1900)
Argentina
jdiaz@info.unlp.edu.ar
<http://linti.unlp.edu.ar/articulo/2008/10/29/jdiaz>

Claudia M. Banchoff Tzancoff
Calle 50 y 120. 2do. Piso
La Plata (1900)
Argentina
cbanchoff@info.unlp.edu.ar
<http://linti.unlp.edu.ar/articulo/2008/10/29/cbanchoff>

Ana Paola Amadeo
Calle 50 y 120. 2do. Piso
La Plata (1900)
Argentina
pamadeo@info.unlp.edu.ar
<http://linti.unlp.edu.ar/articulo/2008/10/29/pamadeo>

Einar Felipe Lanfranco
Calle 50 y 120. 2do. Piso
La Plata (1900)
Argentina
einar@info.unlp.edu.ar
<http://linti.unlp.edu.ar/articulo/2008/10/29/elanfranco>

Francisco Javier Díaz Licenciado en Matemática Aplicada de la UNLP y Profesor Titular Ordinario de la Facultad de Informática de la UNLP. Es Director General, Científico y Tecnológico del Centro de Procesamiento de la Información (CeSPI) y Director del Laboratorio de Investigación de Nuevas Tecnologías en Informática (LINTI).

Claudia M. Banchoff Tzancoff es Licenciada en Informática de la UNLP y actualmente es Profesora Titular y Secretaria de Extensión de la Facultad de Informática. Trabaja en proyectos relacionados a Software Libre, principalmente en aquellos orientados a Educación y Salud.

Ana Paola Amadeo Es Licenciada en Informática de la UNLP y Profesora Adjunta de la Facultad de Informática. Sus áreas de investigación incluyen a Educación a Distancia e Diseño Centrado en el Usuario.

Einar Lanfranco Es Analista de Computación y Jefe de Trabajos Prácticos de la Facultad de Informática. Trabaja en temas relacionados a Seguridad y Software Libre.