

Moodle: A Simple, Intuitive, Open-Source Course Management System

A Tutorial

Adel Abunawass

Will Lloyd

Edwin Rudolph

{adel,wlloyd,erudolph}@westga.edu

State University of West Georgia
Department of Computer Science
Carrollton, GA 30118

What is Moodle?

Moodle is a free, open-source course management system (CMS).

Initially developed by Martin Dougiamas to address:

- complexity and cost frustrations with commercial CMS products.
- desire to produce a free and open-source CMS alternative.
- research interest in social constructivist learning pedagogy and e-learning [2].
- needs and uses of smaller, intimate University-level classes.

These factors continue to drive development, however, with the growth of the user and developer community, additional interests include:

- Scalability to handle larger classes and/or more classes
- Addressing needs of those outside of higher-ed (i.e, K-12, non-profit groups)
- Support for standards (i.e., IMS, SCORM)

Main Features

- Designed to be simple and intuitive – interface is clean and uncluttered, attempts to maintain consistency throughout the system. Tries not to get in the way of learning.
- Course web organization designed to mirror actual structure of course (topic-based, weekly, or social)
- Main components to support distance learning and enhance traditional F2F courses:
 - Assignment submissions
 - Discussion forums
 - On-line quizzes
 - Detailed activity reporting
- Open and understandable interface for customization. Works with various authentication systems and account databases.
- Growing and diverse community of users, administrators, and developers – community support (“Using Moodle” community on project web site [2])
- It’s free!

Why learn yet another CMS?

Commercial CMS products seem to be suffering from “feature bloat” – do we really need more features?

Main advantages:

- Simplicity and intuitiveness: provide essentials and get back to learning and teaching!
- Open to customization
- Zero or very low cost (only hardware, perhaps support personnel depending on size of implementation)

Disadvantages:

- Lacks some features that may be necessary for certain types of courses (i.e., quiz module limited in capabilities)
- Scalability - may not be ready for large-scale implementation
- Still a young system

Learner View

- Course web organization designed to be intuitive:
 - Format of course is mirrored on the course web (weekly, topic-based, or discussion/socially-based)
 - Consistency across courses
 - Assignments and Quizzes appear in context (students do not need to access separate areas to take a quiz or submit an assignment) as well as by module
- Students can focus on the current topic or week by hiding other content
- Course “participants” are easily found, each participant can update their own profile – enhances sense of community online

Instructor View

- Course web layout mirrors the format of the course:
 - Topic-based: not necessarily based on dates
 - Weekly: sections for each week
 - Social: centered around discussion forums
- Course materials are created as “Resources.” Some of the modules that are available include:
 - Discussion forums (supports attachments and e-mail copies)
 - Assignments – both on-line (file submission) and off-line (just for grade/feedback)
 - Links to uploaded materials as well as to external sites

Instructor View

Quizzes

- Supports M/C, T/F, short answer, matching formats
- Questions can be organized into categories (categories may be “published” for use in other courses)
- Random question selection
- Limited support for question file import (currently only Blackboard)
- Drawback: No manually graded questions (i.e., essay)

Instructor View

Administrative tools

- Students add themselves to the course using an instructor-provided “enrollment key” (latest version allows for enrollment by instructor)
- Grades from various assessment tools are recorded and summed, can be downloaded into text or Excel (no sophisticated grade book tool or custom formulas)
- Detailed activity reporting
- “Login as” feature allows the instructor to temporarily log in as a student in their course (maintains instructor login)
- Standard file manager for adding files, provides zip/unzip capability

System Administration: Front-end

- Front-end administration is very straightforward and requires little additional knowledge beyond regular use of Moodle
- Administrator has full access to all courses
- Multiple administrators supported
- Course creator role can be delegated without giving full admin access
- Nice Web-based database admin tool for working with the database (phpMyAdmin) if needed

System Administration: Back-end

- Would involve installation of Moodle, database, and Web server as well as maintenance
- Need to be familiar with basic system administration (can be either on UNIX or Windows) and be familiar with running and working with:
 - Apache
 - MySQL
 - PHP
- “Using Moodle” community on the project web site 2 is an excellent resource for installation, setup, configuration, and administration of Moodle.

Technical Requirements: Hardware, Software

- Hardware needs will vary depending upon size of the course web and the level of usage. In general, guidelines for standard web servers would apply (no specific needs for Moodle).

Our configuration: 2x800 MHz Pentium II, 2GB memory, 36GB disk space

Size of the site: Approx. 18 courses, 200 users

- Software requirements:
 - OS: any platform that can run Apache + PHP and MySQL. Linux, Mac OS X, and Windows probably the most commonly used (we use FreeBSD).
 - For Apache, PHP, and MySQL, most platforms provide pre-built binaries so compiling from source is not required.

Customization

- Provides ability to create custom themes for “branding.” Not yet web-based, however, so some experience with HTML, CSS, and PHP is needed.
- Moodle is designed to allow for relatively easy development of additional modules which can simply be inserted into the system and activated through the administrator interface. Source code is easy to read follow.

Example: CS Survey System

Inserted a small script to provide a “hook” into an external survey package (phpESP [3]) so that authentication to the survey tool could be provided by Moodle (transparent to the user).

References

1. UWG Computer Science Course Web.
(<http://courses.cs.westga.edu>).
2. *Moodle - A Free, Open Source Course Management System for Online Learning*. Project web site. (<http://www.moodle.org>).
3. *phpESP: Easy Survey Package*. Project web site.
(<http://phpesp.sourceforge.net>).

Want to try it out?

We are happy to create a test account for you to play around with for a few weeks if you're interested. Just send me an e-mail at erudolph@westga.edu!