
Software Engineering

Moodle Software Requirements Specification For Puget Sound Enhancements

Version 1.0



Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

--	--	--	--

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

Table of Contents

1.	Introduction	6
1.1	Purpose	6
1.2	Scope	6
1.3	Definitions, Acronyms and Abbreviations	6
1.3.1	Actor	6
1.3.2	Course Administrator	6
1.3.3	Courseware System	6
1.3.4	System	6
1.4	References	6
1.4.1	Moodle Requirements Brainstorming.pdf	6
1.5	Overview	7
2.	Overall Description	7
2.1	Assumptions	7
2.2	Product Characteristics	7
2.3	User Characteristics	7
2.3.1	Students	7
2.3.2	Professors	7
2.3.3	System Administrators	8
2.4	Stakeholder Needs	8
3.	Specific Requirements	8
3.1	Multiple File Transfer	8
3.1.1	The system must allow for multiple file upload to be configured on a per-page basis.	9
3.2	Audio Recording	9
3.2.1	The system must allow actors to organize their voice clips into a portfolio.	9
3.2.2	The system must allow actors to download their voice clips in a flexible format.	9
3.2.3	The system must store audio clips in a format conducive to speech.	9
3.2.4	The system must allow actors to delete recorded clips.	10
3.3	Web Feeds	10
3.3.1	The system must allow web feeds to be turned on or off on a per-page basis.	10
3.4	Search	10
3.4.1	An actor must be able to search through pages in a course.	10
3.4.2	The system must display a search box on every page after an actor has logged in.	11
3.5	Gradebook	11
3.5.1	A course administrator must be able to grade an assignment.	11
3.5.2	The system should display grade information to the appropriate actor.	11
3.5.3	The system should maintain a grade history.	11
3.6	Social Networking Applications	12
3.6.1	The system must provide a wiki where actors can collaboratively create networks of documents.	12
3.6.2	The system must provide a blog engine.	12

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

3.7	Notifications	12
3.7.1	The system must provide both e-mail and SMS notifications for pages.	12
4.	General System Functional Requirements	12
4.1	Usability	12
4.1.1	As far as possible, the system should provide a simple, responsive interface.	12
4.2	Reliability	13
4.2.1	The system must be backed up on a configurable schedule.	13
4.2.2	The system should be available 24 hours a day, 7 days a week.	13
4.3	Performance	13
4.3.1	The system should support at least 1000 concurrent users.	13
4.4	Supportability	13
4.5	Design Constraints	14
4.6	On-line User Documentation and Help System Requirements	14
4.7	Interfaces	14
4.7.1	User Interfaces	14
4.7.2	Hardware Interfaces	14
4.7.3	Software Interfaces	14
4.7.4	Communications Interfaces	14
4.8	Licensing Requirements	14
4.9	Applicable Standards	14
5.	Supporting Information	14

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

Software Requirements Specification

1. Introduction

1.1 Purpose

The purpose of this document is to capture, in natural language and at a functional level, the description and requirements of a new courseware system for the University of Puget Sound's educational needs. This is a functional description of those features required to address current instructional and educational requirements. Each requirement is accompanied by a short discussion, designed to add the background and framework necessary to explain the functionality. It also describes nonfunctional requirements and other factors necessary to provide a complete and comprehensive description of the requirements for the software.

1.2 Scope

This document deals with the immediate requirements for the development of core functionality currently unavailable in Moodle, the existing courseware system candidate (version TODO: Version of Moodle). The University's enhancements to Moodle will embody several small redesigns for consistency with other university practices. Additionally, the enhancements will take the opportunity to redesign the user interface to reflect a more natural flow of data.

These requirements are based on the assumptions that Moodle will be adopted at the University of Puget Sound and that any enhancements to the system will utilize the existing APIs made available by Moodle.

1.3 Definitions, Acronyms and Abbreviations

1.3.1 Actor

Actor is synonymous with user and is used to describe users who are performing actions in use cases.

1.3.2 Course Administrator

Course administrator is an actor who is acting as a professor or system administrator and who has the ability to manage courses and course pages.

1.3.3 Courseware System

Courseware system is a piece of software meant to help facilitate electronic classroom management and provide for electronic grading, assignment submission, discussion, and other learning tools.

1.3.4 System

System refers to the existing courseware system, a new courseware system, or any modifications being made to Moodle while it is being considered as a replacement for the existing courseware system.

1.4 References

1.4.1 Moodle Requirements Brainstorming.pdf

This document was generated as a foundation for this document and contains the results of stakeholder interviews. This document does not make any further reference to the

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

requirements brainstorming document.

1.5 Overview

This is a working document and, as such, is subject to change. In its initial form, it is incomplete by definition, and will require continuing refinement. Requirements may be modified and additional requirements may be added as development progresses and the system description becomes more distilled.

This document also forms the basis for continuing discussions with stakeholders and sponsors, to enhance and further describe those requirements to be included in future enhancements to Moodle. This information will serve as a framework for the current definition and future evolution of the University of Puget Sound's courseware system.

2. Overall Description

2.1 Assumptions

The University of Puget Sound is currently seeking a new courseware system and is investigating Moodle as an option. We are making an assumption that Moodle is of interest to the University as a viable choice to replace Blackboard.

The goal of this document, and our work, however, is to determine what any courseware system would need to qualify as a viable solution for the University. We also intend to apply this document to Moodle to determine whether we can implement some of the features that Moodle lacks and our stakeholders find to be important. By no means is the University necessarily going to choose Moodle as a courseware system, but we want to use the platform to have a new learning experience and contribute back to a real product nonetheless.

2.2 Product Characteristics

Moodle is courseware system that manages courses, assignments, wiki pages, forums, etc. Our goal is to refine parts of Moodle to provide functionality that improves the overall learning experience of students at the University and helps professors to make students aware of new ways of presenting material.

At present, the University does not believe that Blackboard is providing the functionality that students and faculty need to encourage learning. Our stakeholder feedback was gathered to help us determine the requirements of a replacement courseware system and provide analysis of the feature set in other available solutions.

2.3 User Characteristics

2.3.1 Students

Students are the primary consumers of a courseware system. They are accessing information posted by professors, uploading assignments and project files, and discussing concepts.

2.3.2 Professors

Professors are the primary content administrators of a courseware system. They are uploading files, links, and multimedia, and grading assignments in addition to creating new places for

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

students to discuss and collaborate.

2.3.3 System Administrators

System administrators are primarily responsible for maintaining the courseware system. They contribute minimally to the courses themselves, but spend more time modifying the system's configuration and making appropriate updates.

2.4 Stakeholder Needs

In order to determine what enhancements needed to be made to Moodle, we interviewed system stakeholders (e.g., professors and students) to determine what, if anything, is lacking in our existing courseware solution. Our requirements and supplementary requirements are based on that feedback which is summarized in ***Moodle Requirements Brainstorming.pdf*** (referenced in section 1.4.1).

3. Specific Requirements

The priorities defined below reflect the requirements of the development team. The goal was to define those requirements that describe the core functionality required to support and facilitate diverse methods of teaching at the University of Puget Sound.

Requirements have been categorized according to the following definitions:

Priority 1 (Core): Core functionality currently existing in Moodle/Blackboard.

Priority 2: Functionality not explicitly available today, but considered critical to implementing a courseware system at the University.

Priority 3: Just barely outside the parameters of Priority 2; considered very important to supplement critical functionality

Priority 4: Outside the parameters of Priority 2; considered valuable as a supplement to other functionality, but not critical

The initial production release of the University of Puget Sound's enhancements to Moodle will deliver the Priority 1 (Core) and Priority 2 set of requirements, along with such Priority 3 requirements as can be developed within the project timeframe and budget constraints.

3.1 Multiple File Transfer

The system must be able to capture and manage files where appropriate.

One of the fundamental tasks of the system is to allow actors to upload and manage files on certain pages of the system. These might include assignment submission pages, forum pages, wiki pages, and announcement pages. An actor in the role of course administrator should be able to optionally upload multiple files where he or she finds that such as feature would be useful—such as on a devoted project page.

For this requirement, page refers to any distinct upload point (e.g., an assignment, a wiki page, a forum post, etc.).

Priority: 2

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

3.1.1 *The system must allow for multiple file upload to be configured on a per-page basis.*

Actors in the capacity of course administrator or professor should be able to enable multiple file uploads on a page of his or her choosing. For collaborative projects, it becomes important for actors to be able to upload more than one file to a given page, especially in the case of an assignment submission where the assignment might include multiple parts. Such a construct should be offered to the actor during a page's creation and editing stages.

3.2 **Audio Recording**

The system must be able to capture and organize voice clips that can be used where appropriate.

One of the shortcomings of the current courseware system is that the foreign language department is unable to capture and track voice clips—recorded by students—for oral homework over the course of a student's education.

Priority: 2

3.2.1 *The system must allow actors to organize their voice clips into a portfolio.*

Equal and finite drive space needs to be allotted to each actor in the role of professor or administrator to allow for the storing of voice clips recorded and submitted by all actors in the roles of students. Actors in the role of professor or system administrator should be allowed to partition this space for actors in the roles of students. This can be done one of two ways: 1) the actor can be allowed to preview the voice clip before it is archived and determine whether or not he or she would like that clip to *be* archived, or 2) a newly created voice clip can be automatically archived. The first option would allow the actor to throw out an audio clip that is deemed to be unsuitable or inappropriate for an assignment without having to take it out of an archive of voice clips. The actor should have the option to have the voice clip be archived after a preview has taken place. The second option would automatically archive a voice clip once submitted but would require that the actor interact with the archived file. Thus, if the actor chooses to delete the file, it would need to be deleted from the archive of voice clips.

Maintaining an archive of voice clips makes it possible for comparisons to be made between clips to determine whether an actor is improving his or her language skills over multiple courses. The actor should be able to optionally attach one of his or her archived items to any page where uploads are permitted.

The term *archive* used above may also be referred to as a *portfolio*.

3.2.2 *The system must allow actors to download their voice clips in a flexible format.*

Voice clips should be optionally downloadable in a format such as MPEG Audio (mp3) format, since it is widely supported across different web browsers and operating systems. Voice clips should be downloadable for archival purposes or for the purpose of presenting them to other actors.

3.2.3 *The system must store audio clips in a format conducive to speech.*

Voice clips should be stored in a format optimized for speech since this feature is meant to catalog voice recordings. Recordings should only be converted to MPEG Audio (mp3) format

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

when they are being downloaded; a separate viewer should be provided in a cross-platform language for listening to clips in Moodle.

3.2.4 *The system must allow actors to delete recorded clips.*

Since voice clips should be stored in a portfolio for each actor during his or her existence in the course management system, he or she should be able to examine his or her portfolio to select and delete clips if he or she chooses to. This allows him or her to manage archived voice clips and to create a cohesive portfolio that actors in the role of professor can use to judge improvement over time.

3.3 **Web Feeds**

The system must be able to display web feeds, such as *RSS 2.0* feeds, for all pages of the course management system where elements such as forum posts, assignment postings, announcements, wiki alterations and other blocks of information that an actor may view are added. Actors in the role of course administrators should have the ability to determine which of these components have feeds and to whom these feeds should pertain.

If actors are unable to be notified in some way of new content in Moodle, they are unable to efficiently keep track of a course's current affairs. For this requirement, page refers to any distinct block (e.g., assignments, wikis, forum threads, etc.).

Priority: 1

3.3.1 *The system must allow web feeds to be turned on or off on a per-page basis.*

Actors in the course administrator role may not feel the need to provide web feeds for every page in their course. During the creation and editing of a page, that actor should be able to toggle web feed availability.

3.4 **Search**

The system must be able to use search functionality as a way to navigate Moodle pages instead of using hierarchical links.

Moodle can be difficult to navigate and requires too many clicks to be efficiently used. Too many steps to complete basic actions—such as submitting an assignment—lead to frustration on the part of the actor. A search utility enables actors to find what they are looking for quickly in addition to having a hierarchical approach to finding functions of the course management system. For this requirement, page refers to any distinct page, such as an assignment, a wiki page, or a forum post.

Priority: 2

3.4.1 *An actor must be able to search through pages in a course.*

Actors should be able to search for pages at the course level, since most actors will be searching for material in a given course. The search feature should be present in two ways.

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

First, it should be present in the form of a search box and a “Search” button on the top or bottom of every page. More information on this topic is given in sub-section 3.4.2 below.

Second, a single page should be dedicated specifically to the task of searching the course management system. In addition to the basic search field, this page should allow an actor to filter out certain results (such as forum posts or grades, etc.) and search specific sections (or components) of the course management system. Results should appear in categories (e.g., Assignments, Wiki Pages, Forum Posts, etc.) based on relevance. In this case, relevance is determined by results that having the highest number terms matching the search terms. When the actor clicks a search result link, they should be taken to the page corresponding to the search result.

In addition, the search page described above might allow for the possibility of searching the world wide web using a standard search engine such as Google.

3.4.2 *The system must display a search box on every page after an actor has logged in.*

Actors should be able to search from any page. If the actor is hierarchically outside of a particular course, each of his or her courses should be searched and results should be grouped by course. If the actor is hierarchically inside of a course, that course should be searched with an option to search all courses.

3.5 **Gradebook**

The system must allow actors in the capacity of course administrator to post grades associated with assignments for a particular actor in the capacity of student. Actors in the role of professor may want to optionally grade assignments online since they are being submitted electronically by actors in the role of student.

Priority: 1

3.5.1 *A course administrator must be able to grade an assignment.*

Course administrators should be able to post grades for an assignment based on the ratio of points earned to points possible. The course administrator should also be able to attach feedback in the form of text or an attachment (as described above). Time stamps should be attached to the last time the grade was modified so that actors may examine when particular grades were submitted and how an actor in the role of student has improved or lessened his performance in the course.

3.5.2 *The system should display grade information to the appropriate actor.*

The system should display grade information such as the grade for each assignment, averages, and overall grade to the actor to whom the grades belong. No other actor should be able to view another’s grades except an actor in the role of course administrator for the course to which the grades belong.

3.5.3 *The system should maintain a grade history.*

The system should maintain a history of grades for a particular assignment if the course administrator changes them. This way, another actor can review the grade information if their

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

score on an assignment has changed and they wish to dispute it. Course administrators can also track how they have changed a score over time using the timestamps associated with each change to the gradebook.

3.6 Social Networking Applications

The social, collaborative components of Moodle are not very robust. There are better, freely-available solutions which should be integrated into Moodle to provide the best functionality possible for all actors without having to use 3rd party services.

Priority: 3

3.6.1 *The system must provide a wiki where actors can collaboratively create networks of documents.*

A freely available wiki should provide a simple markup language that actors can use to style their input, as well as link to other pages in the wiki. More importantly, having the wiki inside Moodle reduces confusion and allows for the integration of notifications and logins.

3.6.2 *The system must provide a blog engine.*

A freely available blog engine should provide all modern blog functionality, such as tagging, drafting, and comments. These blogs should share authentication and notification with Moodle.

3.7 Notifications

Currently, there is no system that allows actors to receive SMS or email notifications of changes to Moodle pages (such as assignments or announcements).

Priority: 3

3.7.1 *The system must provide both e-mail and SMS notifications for pages.*

For this requirement, page refers to any distinct page (e.g., an assignment, a wiki page, a forum post, etc.). When a page is created, the actor in the role of course administrator should be able to toggle whether notifications are turned on. By default, they should be turned on for announcements. If notifications are turned off, actors in the capacity of student should be able to subscribe to notifications.

Students should be able to select in their preferences whether notifications are sent to their mobile telephone (via SMS) or e-mail account. They should also be able to manage their notification subscriptions (for example, remove themselves from notifications).

4. General System Functional Requirements

4.1 Usability

4.1.1 *As far as possible, the system should provide a simple, responsive interface.*

Although any courseware solution may be composed of diverse systems, applications, and processes, the underlying architecture should be transparent to the administrators. The system should be configurable from a single source at a central administrative position, and should provide a central, easy-to-use interface that will allow administrators to configure the user interface and features in a way that reduces page clutter.

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

A system will be considered to meet this requirement if it has a single administrative interface rather than individual links for editing each page. Furthermore, this interface must allow administrators to easily change themes and other setting that affect page layout across the entire courseware system.

Priority: 3

4.2 Reliability

4.2.1 The system must be backed up on a configurable schedule.

Back-up requirements will need to be determined, based on individual components of the system. The system should be backed up with a frequency that ensures system and course data is protected. Since assignments will be submitted via the system, it should be backed up on a nightly basis, with options for weekly, off-site backup when necessary.

The system must have the ability and capacity to restore back-up data within six hours so that the system is never offline for an inordinate period of time.

Priority: 2

4.2.2 The system should be available 24 hours a day, 7 days a week.

This statement provides a general sense of system availability. It is not intended to demand the system maintain reliability, or to require the system to be highly available. It should not exclude maintenance windows, or scheduled downtime. It is only intended to convey the expectation that our customers should have access to the system during off-hours as well as business hours. 99% up-time should be considered sufficient to meet this requirement.

Priority: 1

4.3 Performance

4.3.1 The system should support at least 1000 concurrent users.

This statement provides a general sense of reliability when the system is under load. It is important that a substantial number of actors be able to access the system at the same time, since a courseware system is important to the courses that employ it. The times when the system will be under the most stress are likely during midterm and finals weeks. Therefore, it must be able to handle at least 1,000 concurrent users.

Priority: 1

4.4 Supportability

4.4.1 The system must be maintainable without substantial modification.

Due to limited staff in the Office of Information Services, it is important that the system be mostly self-sustaining. This will reduce the number of FTE hours spent maintaining the system and simplify maintenance tasks.

Moodle	Version: 1.4
Software Requirements Specification	Date: 10/25/2007

Priority: 3

4.5 Design Constraints

4.6 On-line User Documentation and Help System Requirements

4.6.1 Relevant, online documentation for users should be available on each page.

Users must have easy access to help while interacting with the system. Adequate user documentation should be provided to minimize the number of calls to the Help Desk about problems with the system. Modifications should be reported via the main page to inform actors of unexpected changes. This electronic documentation should be supplemented with phone and on-site support provided by the Office of Information Services.

Priority: 3

4.6.2 System administrators must have access to comprehensive, searchable documentation.

A comprehensive database of maintenance tasks and help files should be compiled to make the courseware system easier to maintain from an IT staff point-of-view. Search results should be displayed based on relevance.

This documentation must cover all procedures necessary for regular maintenance with links to additional information, all common errors, and links or documentation for advanced troubleshooting.

Priority: 3

4.7 Interfaces

4.7.1 User Interfaces

4.7.2 Hardware Interfaces

4.7.3 Software Interfaces

4.7.4 Communications Interfaces

4.8 Licensing Requirements

4.9 Applicable Standards

5. Supporting Information