

Please note – this document is NOT an RFP or legally binding, and is subject to change. Only officially issued RFP documents that may follow the consultation process will be considered official. Proponents should NOT base any formal proposals on this document. This document is for discussion purposes only.

SAMPLE FUNCTIONAL REQUIREMENTS AS PART OF A SAMPLE LME RFP

Table of Contents:

- DESCRIPTION OF CORE LME GENERAL FUNCTIONALITY
- PROCESS
- TECHNICAL SPECIFICATIONS & COMPLIANCE
- APPLICATION ADMINISTRATION & CONTROLS
- USERS, ROLES, CONTENT
- ASSESSMENT SUPPORT
- COMMUNICATION TOOLS

DESCRIPTION OF CORE LEARNING MANAGEMENT ENGINE (LME) GENERAL FUNCTIONALITY

The University is seeking to renew and enhance the e-learning systems and services it makes available to all of our students, Instructors and staff

This Request for Proposal (RFP) is directed to innovative, solution-oriented suppliers who can help us enable a secure and privacy-conscious environment to our end-users

Specifically, we are looking for a Core LME.

We are also interested in considering other tools or systems that may add capacity not readily available in a Core LME, and we have a strong interest in evaluating cloud versus locally hosted options and open source versus proprietary solutions However, any new LME system must offer:

- Ease of use
- Full support for mobile devices including iOS and Android based tablets and smartphones;
- Full support for all modern browsers including but not limited to Chrome, Safari, Explorer, and Firefox running on multiple operating systems
- Excellent performance and seamless scalability from small courses of a few students to massive courses as needed
- Effective and easy-to-use management of multiple sections within a single course shell or similar way to manage multiple course sections
- Learning analytics
- Social networking and sharing
- Excellent accessibility - must not break screen readers on multiple platforms
- Ability to leverage tools in our existing environment or proposed superior tools
- e-text book and course material plugin capacity

PROCESS

This RFP process* includes a pre-selection step where the review committee as part of a short list will select no more than four (4) Core LME systems.

Other third party systems or service add-ons to the Core LME *may* be proposed

In addition to being part of the evaluation for this RFP*, a Demonstration Course for those selected shortlisted product(s) will be provided to *The University* community for further review and community feedback. While the committee has a strong interest in ensuring the pre-selected products include locally and externally hosted options as well as open source and proprietary options, the demonstration course will not be locally hosted. Systems that cannot be made available to everyone in *The University* community who is interested will not be selected.

When the community review of the shortlisted product(s) is complete the committee will evaluate feedback from the community along with their initial evaluation and make a final selection.

** Reminder: This is not an actual RFP document – it is only a sample for discussion purposes*

Requirements and Specifications

The following requirements list should be answered with a Yes or No indicating if your Core LME has the listed requirement, with the opportunity to provide further details where needed for any of the questions.

The University is asking solely about the functionality in the Core LME product. Only answer “Yes” if the functionality is available in the core product offering. If the functionality is achieved with 3rd party add-ons, please answer “No” but include the additional information with a whole language description.

TECHNICAL SPECIFICATIONS & COMPLIANCE

Platforms

- Your system must support commonly utilized Web Browsers, including current versions of Microsoft Internet Explorer, Apple Safari, Mozilla Firefox, and Google Chrome.
- Your system must support the latest commonly used platforms including Windows, Mac OS X, and Linux.
- To avoid many issues with Java installations on client's computers, we ask that your system not use client side Java plugins Can you comply with this?
- Does your system have support for multiple languages?

Mobile Access

- Ability to access full interface on the native browsers of mobile devices.
- Provide mobile apps for common mobile devices (please indicate the specific mobile operating systems on which you have produced apps).
- Provide mobile access free of any charges for students and Instructors.
- Provide mobile access free of any additional charges for the institution.
- Please indicate which tools are accessible via your mobile apps (not the native browsers).
- Provide ability for users to sync some areas of a course in the app for offline viewing (Please describe).
- Provide ability for users to complete some tasks in the app when offline, and sync back to the course when online (Please describe).
- Ability to quickly switch between courses/organizations in the mobile app.

Offline Access

- Does your Core LME allow for working offline?
- If so, what features are supported for working offline? If so, will synchronization occur automatically?

Standards and Compliance

- Your system must be able to support required volumes of class sizes of five thousand active students or more and organizations of one hundred thousand.
- Provide the ability to scale the implementation for increased usage, such that more storage and users can be added.
- Identify any limitation from a student or course perspective (daily usage, quota restrictions).
- System works/compatible with *The University's* technology infrastructure including network operating systems, single sign-on solution, technology stacks and other infrastructure requirements.
- What components of the system's source code can be modified?
- The solution is written in an industry-standard language.
- Meets the IMS Interoperability standards (<http://www.wimsglobal.org/specificationshtml>)
- Meets the latest LTI Specifications (<http://www.wimsglobal.org/lti/indexhtml>)
- Ability to import SCORM or AICC packages (content and assessments), such as those created by Adobe Captivate or Articulate (but not limited to those).
- Ability to export SCORM or AICC packages (content and assessments), such as to Adobe Captivate or Articulate (but not limited to those).

Application Integration

- Supports integration with other Systems or 3rd Party Applications, i.e. supports open standards, including LTI, and provides a rich API. Describe.
- Supports interoperability with other Systems or 3rd Party Applications. Describe.
- Supports web services, APIs, LTIs, real-time synchronization, and batch processes. Describe.
- Can you guarantee the stability and reliability of the API and LTI? Describe how changes are managed.
- Manage access to the API and LTI. Please describe.
- Provides an SLA for the API or LTI component for integration with third-party software.
- Provide API to pull announcements, what's new, list of courses for users, etc. for use via other *The University* systems.
- Provides full integration with Microsoft Office365, i.e. not just SSO integration, allows in-line viewing, editing, and real-time collaborating to and from O365 tools.
- Provide API to enable real-time integrations (i.e. simulations), for use in synchronous tools such as virtual meetings
- Provide the ability to integrate with specific resources from within the Core LME (example from viewing grades in *The University's* SIS to viewing grades in the Core LME).
- Provide APIs that support server side Java function calls (not remote procedure calls via REST or Web Services)
- If an operation cannot be performed through the API, what mechanisms are in place for *The University* to achieve the functionality?
- Have you published your APIs? Have you provided API documentation with your proposal?

Integration with Identity Management System

- The Core LME must be integrated with *The University's* Identity Management system. It is imperative that account management including the creation,

modification, disabling, and deleting of accounts, roles, user groups, programs of study, course shells, and course shell registrations be managed by *The University's* Identity Management System. We would also like the capability to structure our courses into a hierarchy that can be grouped into course, program/department, Instructors, etc.

- Do you have real-time APIs in your Core LME to provision user accounts including create, re-activate, deactivate and delete user accounts?
- Do you have batch process to provision user accounts including create, re- activate, deactivate and delete user accounts?
- Do you have secure protocols in place for data transfer between our Identity Management System and the Core LME for both real-time API and batch process?
- Does your system automatically purge deleted accounts including user related content? If not automatically, are there APIs available to do the purge?
- Do you have APIs to support username changes?
- What is used to protect stored user, administrator, and system passwords? Cryptographic hash function?
- *The University's* Identity and resource management system, provisions courses in the Core LME. *The University* would like to structure our courses into a hierarchy that can be grouped into course, program/department, faculty, etc. Do you have a real-time API for *The University* to provision the hierarchy accordingly? Example: course creation, course enrollment (Instructors, students, TAs, etc.), deactivation, deletion, reactivation, and copying of courses?

Authentication

- Is your system CAS (Central Authentication Services) enabled? (<http://www.jasig.org/cas>)
- Does your system support SAML?
- In terms of your service's web interface, is the logout customizable so that *The University's* systems are notified in real-time that the user has logged out? For example via a redirect to CAS or via other methods?
- Can we customize the logout process? For example, if an end user logs out of *The University's* portal, CAS, or other system and we want to also log the end user out of the Core LME
- Can the logout button in your services be hidden if the Core LME is being used within *The University's* portal?
- Ability to disable the local login.
- Do you support CAS web authentication for your mobile app?
- Provide the ability to configure the session timeouts.
- Ability to dynamically authorize permissions by role to a user upon authentication.

Accessibility Compliance

- Can your company document the conformance of your system to the World Wide Web Consortium (WC) WCAG guidelines Success Criteria Level AA? Indicate Yes / No / N/A and explain Refer to: <http://www.w3.org/TR/WCAG/> Please provide your company's accessibility statement
- If your answer is No: a) describe the undue burden, i.e., the significant difficulty or expense incurred, in order to comply with the WCAG standard If the monetary expense is a basis for the undue burden, explain the costs and how they were estimated Be sure to quantify the effort in time and money to make the proposed system/resource compliant; b) Describe the insufficient resources available to the office or program for which the product(s) or service(s) is/are being acquired If technical difficulty is claimed, describe and document such difficulties
- If you answer is Yes, describe the methodology used to ascertain conformance

- Perceivable: Does your Core LME provide text alternatives for any non-text content and components so that it can be changed into other forms people need, such as large print, braille, speech, symbols, or simpler language?
- Perceivable: Does your Core LME provide alternatives for time-based media?
- Perceivable: Can your Core LME create content that can be presented in different ways (for example simpler layout) without losing information or structure?
- Perceivable: Does your Core LME make it easier for users to see and hear content including separating foreground from background?
- Operable: Does your Core LME make all functionality available from a keyboard?
- Operable: Does your Core LME make all functionality available from a keyboard?
- Operable: Does your Core LME prevent the design of content in a way that is known to cause seizures?
- Operable: Does your Core LME provide ways to help users navigate, find content, and determine where they are?
- Understandable: Does your Core LME design (aside from content being provided/uploaded/created by users) make text content readable and understandable?
- Understandable: Does your Core LME design/interface appear and operate in predictable ways?
- Understandable: Does your Core LME help users avoid and correct mistakes?
- Robust: Does your Core LME maximize compatibility with current and future user agents, including assistive technologies?
- Please describe in detail how you ensure accessibility is built into your proposed solution from the standpoint of your product release cycle
- Does your company have the relevant experience of applying the principles of ICT accessibility that create solutions that are accessible to the widest number of users, including users with disabilities?

Information Security and Privacy Policies and Practices

- Do you have a current and formalized information security policy? Is it published and publicly accessible? Please provide for review
- Do you have a current and formalized privacy, data protection, or related policy that protects client data, specifically personal information? Is it published and publicly accessible? Please provide for review
- Do you implement recognized information security standards and practices in your application development and operations? Please list and describe
- Do you subscribe to international privacy standards? Please list them
- Do you have procedures and controls in place to protect *The University's* and your own organizational assets, including information, software and hardware? How are they implemented and monitored/enforced?
- What are the terms of your SLA? Make sure to provide SLA details for the following: uptime, security incident resolution procedures, outages and business continuity management Please specify timeliness of client notification and handling of a security incident, measures that may be taken for incident containment, and cooperation in incident investigation and resolution
- Would you permit *The University* to run its own vulnerability scans against your product?
- Are you subject to regular security audits or assessments (preferably conducted by a third party), such as Threat and Risk Assessments (TRAs), vulnerability assessments, code review, or service/operational audits such as SAS, SSAE, Uptime Institute Tier certification? What were the audit findings? Have the audit findings been mitigated in a timely and effective manner? How were the mitigation actions validated? Can you provide any evidence or attestation of compliance or certifications attained?

- *The University* is subject to Ontario's Freedom of Information and Protection of Privacy Act ("FIPPA"), which contains requirements about protecting clients' personal information, providing notice about the collection, use, and disclosure of personal information Please explain how you have supported other Post Secondary clients' ability to comply with Privacy and Data Protection Legislation (and other regulatory requirements) Examples from Ontario would be preferable
- Does your system utilize any third party products or subcontract to third party service providers? If yes, what are the liabilities and responsibilities of the third party as they impact *The University* in the event of an incident or breach
- How do you ensure the integrity of your software and information against malware and other harmful, unauthorized mobile code?

Monitoring and System Logging

- For a hosted solution, what are your safeguards to determine whether there has been any compromise of the relevant assets? (e.g. loss or modification of information, software and/or hardware)
- Are system events logged with sufficient information to ensure traceability to a unique individual or system?
- For a hosted solution what is the scope of your logging and monitoring
- System logs all actions/creates audit trail on documents and preserves user work history even after user is removed
- Are system logs visible and searchable in administrative GUI
- Please describe how your logging infrastructure adequately protected against unauthorized tampering

Network Security Management

- How is data in transit secured over untrusted networks such as the Internet?
- If hosted locally at *The University*, what network requirements for hosting this at *The University*, e.g. firewall changes, ports opened, etc.
- Describe whatever OSI layer protocols used in your solution, particularly to secure data transfer Include protocol names, versions and identify the uses

APPLICATION ADMINISTRATION & CONTROLS

Administrative Meta Controls

- Provide the ability to configure the interface to *The University* standards (colour schemes, logos, headers/footers)
- Provide the ability for administrators to post and edit system-wide announcements
- Granular admin configurations for tools/features (i.e. so that we can turn on or off and hide features we do not use)
- Ability for system administrators to define allowable file types for any kind of file upload submissions (i.e. assignments)

General Interface

- Provide proper validation and error handling when uploading files (i.e. not allow files with names containing non-alphanumeric characters) Should be consistent throughout the system, anywhere a file may be uploaded
- Ability for system administrator to display a prominent notice on the homepage of each course
- Ability for system administrator to enforce a pop-up notice the first time a user enters a course shell User must 'agree' to continue
- When uploading a file, anywhere in the course and by any user, enforce a step-through process where users have to select what kind of copyright permission they

have for the file The system needs to log what was selected and be able to retrieve that information for reporting purposes

- Provide a descriptive warning when an item is to be deleted (not just 'are you sure', but clearly indicating what is to be deleted)
- Ability for Instructors to simulate and see a student view of their course

User Profiles and Privacy

- According to the Privacy by Design Principles, if a user profile feature is available in a system, the default setting of the profile is "Private." Can the System Administrator accomplish this?
- Does your Core LME allow users to control and configure the visibility of their profiles, including what to share or not share? How granular are the access controls?

User Management

- Ability to provision users to the system based on their institutional role in *The University's* identity and resource management system (integration with *The University's* Identity Management System (encompassing systems such as SAS and HR))
- Provide the ability to assign course administrators to a selected group of courses based on institutional role and/or course category
- Provide the ability to manage user profiles, where roles other than the ones in *The University's* Identity Management System need to be assigned in the Core LME
- Provide the ability to assign storage space parameters based on user role
- Provide the ability to modify/update/restrict specific users storage space
- Ability for System Administrator to search and edit users in the system based on personal details (user name, Last/First Name, student number)
- Ability for Admin to search and edit users in the system based on system roles

Role Management

- Provide the ability to define system roles and course roles
- Provide the ability to assign privileges to course roles (i.e. Instructors should have access to all system-defined course areas and tools, students should have read-only access to content and student tools and not to the Grade book, guests should have access only to areas/tools that instructor sets, etc.)
- Provide the ability to assign privileges to system roles (i.e. System Administrator should have access to all system settings, Help desk role should have limited access to courses, Course Administrator should have access to specific course category - Term, Department, and a system admin role for a subset of courses, etc.)

Course Enrollments / Shell Membership

- Ability to add users to courses based on their enrollment information in *The University's* Identity Management System (encompassing systems such as SAS and HR)
- Ability to add one or more sections to a course (combined sections)
- Ability to automatically identify the section each student belongs to within the course shell based on their enrollment information in *The University's* Identity Management System
- Ability to disable/remove users from courses based on their enrollment information in *The University's* Identity Management System (encompassing systems such as SAS and HR)
- Ability to update users access to courses (drop from a course and re-add to another) based on their enrollment information in *The University's* Identity Management System (encompassing systems such as SAS and HR) (section number)

- The dropped/disabled students should not have access to course content, tools, receive emails, etc.
- Ability for Admin to manage user registrations (manually add/remove courses to a users accounts)
- Ability for admin and Instructors to add course users outside of the course registration (add/drop students, add/remove guests, etc.)
- Ability for admin to enable and disable self-enrollment for a subset of courses

Course Shell Creation and Management

- Ability for system administrator to control who can create new courses in the Core LME
- Ability for admins to create courses while copying content and tool settings from previous courses for a specific term
- Ability for admin to create courses based on a selected template (department/school template, etc.) for a specific term
- Ability for admin to batch create courses (new or copies)
- Ability for admin to delete courses in batches
- Ability for admin make multiple courses visible/not visible to students
- Ability for admin to disable Instructors' access to specified courses at the end of term
- Ability for admin or Instructors to copy specific course content/areas/tools from a course to another
- Ability for admin or Instructors to export a course package that can include the course content, tools, assessments, etc. (but not the student data)
- Ability for admin or Instructors to import a package exported from a course
- Ability to create courses that can be open to non-*The University* users
- Ability to keep old courses in the system for a specified number of years (in accordance with *The University* retention policy) and be able to allow Instructors to access them for various reasons (appeals, reviews, etc.)

USERS, ROLES, CONTENT

Manage Users

- Ability for an instructor to identify students (i.e. at risk students, students who missed a deadline, etc.)
- Ability to handle large classes of five thousand active students and organizations of one hundred thousand members
- List your maximum number of active users in a shell that can be handled without impact on system performance
- Ability to have multiple Instructors sharing a course
- Ability to assign specific Instructors or TAs to specific student sections within the same course shell
- Ability for system to add users with various defined roles (levels of access) to courses: Instructors, Co-Instructors, Teaching Assistants (TA), Graders, Students
- Ability to export completion records for compliance training and performance, competency and career development to the HR system (students/Instructors can download assessment results/reports, certificates of completion, etc.)
- Can admins restrict Instructors from changing their role in the Core LME

Groups

- Ability for Instructors to create a set of self-enroll groups, from which a student may only self-enroll in one

- Instructor can manually add group members from course membership
- Instructor can allow students to self-enroll in predefined groups, where group size may be restricted
- Instructor can instruct the tool to randomly add group members based on criteria (i.e. determine the number of groups, or number of members in each group)
- Instructor can create a number of groups based on the same criteria
- Ability for instructor to, at a glance, view group membership (count) by group
- Ability for instructor to, at a glance, view a list of students to see which groups they are enrolled in
- Ability for instructor to determine whether students are allowed to leave a group and join another, in the case of self-enroll groups
- Sign-up sheets (for groups, topics, presentation time, etc.) as a means of joining a self-enroll group
- Provide private group versions of any collaborative and communication tools (i.e. email, discussion, virtual meeting, wiki, blog)
- Ability for group members to exchange files through the proposed system
- Dynamic group membership based on section number
- Dynamic group membership based on other criteria Elaborate in the details
- Ability for instructor to control the visibility of the group membership to the entire course
- Smart management of users, e.g. handles students who dropped and registered late as they come, does not include dropped students Ability to email/notify instructor when group members drop the course
- Subgroups of groups can be created
- Course content and tools can be easily assigned to groups, and can be customized for each group (e.g., different due dates on assignments)
- Ability to assign a folder in Google Docs for the use of the group members, not visible to other students in the course shell
- Ability, for group members, to share their work with the entire class
- Provide the capability for student group activities (discussions, etc.) to be private (not viewable by the instructor - instructor initiated)

Manage Course Content

- Provide the ability for a user to export a course (content, tests, tools) and import the resulting package into other courses
- Ability to post/release content/assessments/tools/communication for specific student sections
- Ability to access history of all changes in the course (communications, content, assessments) with data for each change (who, when)
- Ability to roll individual items back to a previous version (for example, specific piece of content)
- Ability to track student completion/test results for all courses in which each student is enrolled
- What authoring tools does the system support? Ability for Instructors to edit content in the system (vs. downloading a file, editing it on the computer and re-uploading it)
- Does the system support the ability to roll back to a previous version of content?

Manage Course Settings

- Ability to make a course available/not available to students
- Ability to categorize and group courses per Terms/Years/Departments
- Ability to change settings per group of courses, based on

- Term/Year/Departments (i.e. select course category 'Fall ' and make all courses unavailable to users)

Manage Course Tools

- Ability for Admin to set which tools are visible in a course by default
- Ability for Instructors to set which tools are visible/used in a course

End-Of-Term

- Ability for Instructors to download/export the course content, student reports, assessment submissions, grades, communications that they can save on their computer

Adaptive Release / Conditional Release / Release Criteria

(In the following section, the term "item" is used to refer to ANY area of the shell that allows adaptive release)

- Provide the ability to selectively release items to students based on multiple criteria
- Ability to define how long an item is available for a student to access (start and end date)
- Provide the ability to selectively release items to an individual or select individual(s)
- Provide the ability to selectively release items based on membership in a group
- Provide the ability to selectively release items based on grade book data
- Provide the ability to selectively release items only to users who have attempted a specified assessment (quiz/survey/assignment)
- Provide the ability to selectively release items only to users who have reviewed a certain piece of content or area of the course shell
- Provide ability to mass update release dates/times (for example, when a shell is copied and the dates need to be updated to the new term's dates)
- Provide ability to set DIFFERENT release criteria to different people / groups of people

ASSESSMENT SUPPORT

Grading Tool

- Provide a centralized grade book for each course shell
- Ability for Instructor to create grade book columns
- Instructor may create columns of different data type, such as numeric, alphanumeric, percentage, letter grade, complete/incomplete etc.
- Ability to categorize columns (i.e. assignment, midterm, final grade, quiz)
- Ability for Instructor to create categories
- Auto-save of each grade as it is entered into a cell (As opposed to editing the whole column and saving at the end)
- Ability for Instructor to move a column in three clicks or less (when already in the grade book)
- Ability for Instructor to select column visibility for their own view of the grade book
- Ability for Instructor to select column visibility for the students
- Ability for Instructor to select column visibility for selected students and groups
- When a column is created (either manually or by another tool), the students should not be able to see the column by default The Instructor should have to make it visible
- Ability for Instructor to edit all column settings
- Ability for Instructor to duplicate a column and its settings within the grade book
- Ability for Instructor to remove columns

- Ability for Instructor to rename columns
- Ability for Instructor to select multiple columns at once, to apply a change to all
- Ability for Instructor to batch remove columns (as opposed to one-at-a-time)
- Ability for Instructor to batch change visibility of columns
- Ability for Instructor to apply other changes to multiple columns at once (such as rename)
- Ability for Instructor to batch create columns
- Extra level of security, presented via dialog box, to prevent accidental deletion of columns that contain grades, assignment submissions, etc.
- Ability to enter a grade for a student in or fewer clicks (from in the grade book)
- Provide the ability to override grades for assessments
- Ability to display dynamic column numbers
- Row count (simple interface to count the number of students currently visible)
- Print function (not just the browser print, but a fully compiled spreadsheet that can be printed without downloading first)
- Ability to customize print (by selecting which columns to print, print filtered results, etc.)
- Ability for Instructors to track all changes to the grade book (add/remove column, add/remove grade, change of settings, etc.), including who made the change and when This is from the GUI, not a log search
- Instructor can customize the display of columns (including column width, text alignment, number of decimals to display, etc.)
- Ability for Instructors to enter grades in the Grade book while not connected to the Internet (offline)
- Ability for Instructors to enter grades in the Grade book while logged in to the system from a Mobile device
- Ability to restrict TA's access to grade book via custom criteria (i.e. filter by specific to limit TA to their own section)

Feedback

- Ability to add text-based feedback to a grade in three or fewer clicks (from in the grade book)
- Ability to provide a variety of feedback types (e.g., text, audio, video)
- Ability to choose to display text feedback on the spreadsheet itself (no additional clicks needed to view it)

Searching, Sorting & Filtering

- Ability to identify students who have dropped the course
- Ability to automatically/dynamically filter out students who have dropped the course (i.e. are "inactive" in the course)
- Ability to filter grade book data by custom criteria (i.e. by sections of students, by student performance on tests, by column category, groups etc.)
- Ability to search the entire grade center by keyword
- Ability to filter search results by selecting criteria within the search interface (filter by student name, filter by column name, column type)
- Student visibility should be dynamically set, using filtering by criteria, with no ability to manually hide individual students
- Ability to select a single column and view only it and all associated details on one page (i.e. column settings, student list, grades, number of attempts feedback, attached files)
- Ability to sort by any grade column
- Ability to select a secondary sort key (i.e. sort by last name within each section)
- Calculations & Formulas
- Ability to create a column that weighs other specified columns

- Ability to create a column that averages other specified columns
- Ability to create a column that adds other specified columns together
- Ability to calculate by column category or grouping (i.e. quizzes, weekly sessions)
- Ability to drop a grade in the calculation, by criteria (i.e. drop lowest, drop highest, drop last)
- Ability to calculate students' scores (average, median, total, etc.) for individual columns
- Ability to include Bonus marks to a final grade calculation
- Ability to use calculated columns, with calculation formula customized by instructor

Integration

- Provide the ability to integrate with third-party assessment tools, including the following: Turnitin, WebAssign, McGraw-Hill Connect, Clickers, etc.
- Ability for instructor to assess and grade discussion participation and postings (hooks into grade book tool, aggregates user's posts and gives basic quantitative stats)
- Ability for students, teaching assistants, to rate other students on discussion postings (connects with the grading tool for assessment purposes)
- Provide ability for Instructors to transfer the final calculated grades from Grade book to *The University's* student information system
- Provide the ability to transfer grades from a test response system (such as *The University's* 'Scantron' based test response system) to the Grade book
- Ability to assign a student with an "incomplete" or other status, and submit this to the student information system

Advanced

- Ability to create and edit grade schemas / ranges (i.e. to implement letter grades)
- Ability to allow Instructors to customize existing Grade Schemas / Letter Grades
- Ability to lock the grade schemas system wide (so Instructors can not edit)
- ability to record assessment results from standard integration packages (such as SCORM and AICC)
- When removing a column tied to other tool (assignment, clickers, etc.) and vice versa, the Grade book is synchronized

Import/Export

- Ability for Instructor to export any/all content/columns/comments from grade book to spreadsheet file (like .csv, .xls, .xlsx) for offline viewing
- Ability for Instructor to import data into the grade book from a spreadsheet file (such as .csv, .xls, .xlsx) Data may be imported into an existing column, or may represent a new column
- Ability to selectively choose what to import or export
- Ability to import a comments column (paragraph format) from a spreadsheet

Rubrics

- Provide the ability for Instructors, Students and TAs to use rubrics to grade anything that can be assessed
- Ability to create rubrics
- Ability to edit rubrics and give a warning if that rubric is in use
- Ability to delete rubrics and give a warning if that rubric is in use
- Ability to customize rubric columns and rows
- Ability to customize rubric scoring/weighting
- Ability to choose rubric type (example: percentage, points, text, etc.)
- Provide the ability to use one rubric in multiple course shells
- Provide the ability to copy rubrics from semester to semester
- Provide the ability to copy rubrics from one course to another

- Ability to export and import rubrics

Other Grading Types

- When grading a graded item, (assignments, wiki, blog, and other tools) ability to see details such the name of who submitted, student number, user name, submission date/time, and including the group name and a list of the group members where applicable
- Have option “to grade” various tools, which will then create a column in the grade book For example, discussion forum, blogs, wikis

Assignment Creation

- Capability to create and deploy assignments where student can upload a file as an assignment submission
- Capability to create and deploy Group assignments, where ONE/any group member can submit on behalf of the group, and it assigns the submission (and resulting grade) to all group members
- Ability for instructor to allow any and all group members to submit for any part of an assignment, not just a single "group leader" (multi-part assignment, revisions, resubmissions, multiple attempts)
- Ability for instructor to configure single, defined number, or unlimited submission attempts
- Automatically create column in the Grade book when an assignment is created
- Ability for instructor to allocate points for an assignment
- Ability to create multi-part assignment
- Ability to allow students to submit revisions
- Provide the ability for system admins to change the file size limit on files that are submitted

Assignment Submission

- Capability for students to submit multiple files for one assignment submission attempt
- Ability for Instructors to define submission file type, where students are restricted to only submitting that file type
- Ability for instructor to set an assignment due date and time, and ability to restrict submissions past this date
- Allow Assignment Submissions from Google Drive
- Ability for Instructors to see student submission dates and time for an assignment
- Provide students with receipts for assignment submissions
- Ability for instructor to grant extensions for individual users or the class for assignments

Retrieving Assignments

- Instructor has the ability to batch download/export the assignment submission files
- Instructor has the ability to download individual student's submission files
- Instructor has the ability to batch upload marked-up assignments and release back to students
- When batch downloading assignments, it must be clear to Instructors who submitted and what assignment it is

Assignment Grading

- Allow Instructors to grade assignments
- Allow Instructors to provide feedback on assignments
- Ability to do Inline marking, including annotation, commenting, rubrics, grade interface (like Grade Mark in Turnitin)

- Ability for Instructors to download the feedback they've given students

Analytics Printing

- Ability to print data and graphs

Reporting Test Results

- Granular data results (various aggregation, averages, medians, per question, per test, per category, etc.)
- Ability to generate various reporting formats, e.g. charts, spreadsheets, graphs, etc.
- Ability to export/download student response data (raw and aggregate data)
- Provide the ability to analyze and report test results, and then flag interesting/unexpected/extreme results (i.e. a large number of students answered a specific question)
- Ability to track and log student activities while taking the test (footprints, time on task) For example when students select an answer (i.e. time stamp of answer choice) within a test or survey

Grade Reporting

- Provide the ability to support statistical analysis of assessment results
- The ability to compare the statistics of multiple assessments
- Ability for the instructor to display graphs showing average class marks distributions
- Provide the ability to create standard and customized reports on individual students grades, and also on overall class grades
- Ability to generate reports based on multiple criteria (ex all students that got below % on more than one assessment)
- Does the system provide both pre-testing and post-testing capabilities (including reports that compare pre- and post-test performance)?

Aggregate Data

- Provide detailed student tracking for each course tool (How many discussion posts were read, how long did they spend on each quiz question)
- Provide reporting on individual level activity
- Provide reporting on course level activity
- Provide reporting on program level activity
- Provide detailed reports on student activity (log in, log out, duration, last accessed, IP)
- Ability for system administrators to enable Instructors/students to see who is currently logged into courses
 - Are there alternative customized modules that can be built for tracking activity?
 - The ability to export completion records for compliance training and performance, competency and career development to the HR system (PeopleSoft)
- Provide the ability to keep a student activity history in a course after the student has dropped the course
- Ability for Admin to set a specific/custom default layout for each department/school
- Ability to create not only course shells, but also organizations that may contain , users,

COMMUNICATION TOOLS

Announcements

- Ability to post announcements to an identified group (e.g. a course)
- Ability to post an announcement via mobile app
- Ability to post an announcement by sending an email

- Ability to edit announcement (even after it has been posted)
- Ability to delete announcement (even after it has been posted)
- Ability to create a draft announcement and save it for later editing
- Ability to schedule announcements for release
- Ability to trigger emails to be sent to users under pre-defined conditions
- Ability to lock certain announcements to remain available
- Ability to mark certain announcements as "Important" or "High Priority" with a visual queue (both for creators and viewers of announcements)
- Ability for viewer to view a list of their personally-marked announcements
- Ability to control the order of the announcements
- Allow the copying of announcements to other groups of users (e.g., a second course)
- Allow announcements to be triggered from other tools

Email

- Email Tool (system interface for sending email)
- Ability for Instructor to send an email to the email accounts of any/all users in the course shell (based on shell membership)
- The instructor should always have a way to refer to the full list of recipients on sent emails
- Any user sending an email should receive a copy in their email account
- Ability for Instructor to determine who the students may email (all users in the course, Instructors, TAs, Groups)
- By default, students should only be able to email the Instructor of the course
- Functionality to send an email from within Groups tool in the course, for easy email of a subset of students
- Functionality to send an email from within other tools in the course, for easy email of a subset of students
- By default, prepended subject field with the shell code in order to identify the email and facilitate email filtering
- Ability to attach documents
- Ability to access a history of sent messages (include subject, date/time, recipients, sender)

Instant Messaging

- Ability to do instant messaging one-to-one
- Ability to do group chat (multiple participants) List the max number of users that can participate in a chat simultaneously
- Ability for the user to show/hide his/her presence
- Ability for user to change status to Online, Away, Free to chat, On the phone, Do not disturb
- Ability to block users from contacting you or seeing your presence
- Provides the ability to share files in the IM tool (list any file type restrictions or size limitations?)
- Provide the ability for two-way video in the IM tool List if the video chat requires plug ins
- Ability to record chat text history, configurable by instructor
- Ability to initiate encrypted chat session with another user Chats will be encrypted over tls across the network
- Works with most XMPP/Jabber clients (Windows, OSX, iOS, Android, etc)
- Ability to configure the system so that presence information is hidden for all users by default

Calendar/Scheduling

- Ability to enable a course calendar
- Ability for Instructor to add events to a course calendar
- Ability for students to add their own events, notes, reminders, etc. to a personal calendar (not the course calendar)
- Ability for users to view calendars combined (i.e. all course calendars plus personal calendar)
- Ability for users to view calendars individually
- Ability to select view (such as day, week, month)
- Allow the ability to create recurring calendar entries
- Ability for Instructors to edit calendars individually for multiple course sections (different due dates, etc.) within the same course shell, where students only see the due dates for their own sections
- System has various ways to edit and reschedule events
- Search calendar events by various criteria (keyword, etc.)
- Ability for any user to filter calendar events by type (i.e. due dates, etc.)
- Ability for Instructor to initiate a calendar event from within a course tool that is date-specific (For example: assignment due dates, scheduled chat sessions, quizzes, sign-up sheets)
- Ability for Instructor to initiate a calendar event from within a third-party course tool that is date-specific (For example: assignment due dates, scheduled chat sessions, quizzes, sign-up sheets)
- Ability to set automated reminders for calendar events (i.e. email or pop-up a week prior to due dates, etc.)
- Integration with Google Calendar: Export/sync to Google calendar (list how this is possible in the further details column: IE via API, building block, or even just iCal subscription URL)
- Ability to integrate and sync with other calendars

Notifications

- Ability for sys admin to enable email notifications to users when course shells, organizations, etc. are updated
- Ability for sys admin to specify which areas within shells should be available for notifications (i.e. content, communications, grades)
- Ability for individual users to opt-in to notifications
- Ability for individual users to select which courses or organizations to receive notifications for
- Ability for individual users to select which areas within shells to receive notifications for (i.e. content, communications, grades)
- Ability to send institution-wide notifications to ALL system users, regardless of whether they have opted-in to notifications
- Ability for sys admin to set grouping of notifications (i.e. individual, digest)
- Ability for sys admin to set timing of notifications (i.e. digest email sent at 00:00 am)
- Ability to control the flow of emails, to prevent strain on the system
- Please indicate which of the following areas are included in notifications:
 - Content (New content added, modified)
 - Assignments (new assignment posted)
 - Due date reminders
 - Grades (new grades posted)

Communication Tools (new discussion posts, announcements)

- Activity Stream / What's New (viewed within the Core LME)
- Ability for users to view an aggregated list of course/organization/institution updates

Text Box Editor (WYSIWYG)

- Every part of the system using a text box should use the SAME WYSIWYG editor
- The solution has WYSIWYG editing that allows standard formatting styles (e.g., heading styles, bullet lists, bold, italics, etc.)
- Provide spell check
- Provide equation editor
- Ability to create web links
- Embed Compatible, to provide EASY embedding of content from applicable third-party systems (like YouTube, Vimeo, etc.)
- Ability to attach and/or embed various media from various sources (images, videos, YouTube, etc.)
- Ability to paste from Microsoft Word and retain formatting
- Ability to paste from Microsoft Word without affecting other elements of the page/course (we have experienced MS code breaking elements of the page)
- Ability to edit HTML source

Search

- Provide a course-wide search
- Please indicate the areas included in course-wide search below

SAMPLE ONLY