Application Details

FY18 Innovations in Teaching With Technology Awards: Application Review

Application Title: Content-Rich Interactive Student Evaluation Platform

Review Deadline:

Applicant First Name:

Applicant Last Name:

Co-Applicant (1):

Co-Applicant (2):

Applicant Degree(s):

Email Address:

Phone Number: (

Primary Organization(s):

Primary Appointment Title:

Contact Person's Name:

Contact Person's Email Address:

Contact Person's Phone Number:

Dept. Chair:

What do you intend to do?:

This project will provide instructors with a platform that allows delivery of content-rich evaluation questions in STEM-based courses. This project involves developing and deploying content generation tools, exemplary content-rich templates customized to the engineering curriculum, faculty training materials, and rapid individualized student feedback interfaces. The need to evaluate students on open-ended design questions with potentially infinite correct responses poses a significant logistical challenge as the faculty-to-student ratio within the College of Engineering increases. The College of Engineering has doubled in student enrollment in recent years with a significantly smaller rise in teaching resources. This college growth has led to much larger class sizes and increased burden for creating, administering, and grading evaluative resources (homework, quizzes, and tests). As a consequence, there has been a trend to move away from free response design-driven questions that encourage the integration of multiple concepts where partial credit or numerous correct solutions mimic real-world engineering practice. In response to the more substantial evaluation burden of

larger class sizes, many of the engaging free response multi-part questions have recently migrated towards single topic multiple-choice problems for the efficiency of grading. The proposed platform provides the ability to scale custom design-driven problems to a higher number of students by reducing the evaluation and feedback burden by instructional resources.

In response to charges by the Engineering Faculty to the Teaching and the Technology committees, a pilot project has been initiated over the past nine months to test mechanisms for reducing the burden of faculty in generating, deploying, and grading student evaluations. The pilot project involved using the open source PrarieLearn testing platform for the Computers In Engineering (ENGR:2730) course.

PrairieLearn is an online problem-driven learning system for creating homework and tests. It allows questions to be written using arbitrary HTML/JavaScript, thus enabling compelling questions that can randomize and auto grade themselves. These problem sets can be developed to access client- and server-side libraries that handle tasks such as graphical drawing, symbolic algebra, and student code compilation and execution.

In addition to generating enthusiasm for pursuing the expanded use of the PrairieLearn platform, the initial success of the PrairieLearn pilot project prompted the Engineering Faculty Teaching and Technology committees to investigate studying adequacy and effectiveness of electronic tools and facilities available for offering online courses. These committees must make recommendations to improve the experience of faculty who teach online classes, including the logistics of performing evaluations for remote learners. The PrairieLearn platform may provide a platform for satisfactorily addressing this faculty need.

The initial review of the pilot project identified weaknesses that impede a full adoption of the PrairieLearn tool for several courses. The richness of design-driven problems is only limited by the ability of faculty to program and test them in the HTML/JavaScript languages. The up-front one-time burden results in reusable learning cases. Each learning case can subsequently be for each student for many semesters. The first aim of the proposal provides the resources (a team of student programmers) necessary to remove the impediments that hinder the adoption of PrairieLearn for several identified engineering core courses. The second aim of this project will develop and deploy the technologies necessary for implementing the policies needed to ensure academic integrity while providing the evaluation of students enrolled in online courses.

Specific barriers to success addressed in this proposal include augmentations to the PrairieLearn system needed to align with the University of Iowa teaching objectives.

Under the guidance of faculty, student programmers will instrument representative examples of enhanced STEM relevant problem-driven interactive questions. These example questions are templates that faculty will use for rapidly generating larger question banks. PairieLearn provides custom programmatic instrumentation of graphical interactive queries

and responses that are beyond the capabilities of the Canvas platform. The student programmers will develop training materials (videos, vignettes, written documentation) for new question generation by faculty members, and resources for students on how to use the PrarieLearn environment.

We will provide mechanisms for instantaneous auto-grading of some problem formats. Questions in PrairieLearn can be formulated to provide real-time feedback on partial answers. This project will provide an infrastructure that can scale to provide weekly or daily unique practice examples that re-enforce topics covered in class.

A set of web-based interfaces will be refined to assist with writing interactive problem statements. The exercise question editor and renderer allowing faculty to create and display interactive questions is based on adaptations the Perseus editor developed for the Khan Academy platform.

We will provide an instructor-friendly speed-grading interface for evaluating, assigning partial credit, and provide personalized feedback on open-ended design problems. Generating specialized speed-grading tools address the needs of faculty to review and provide individualized feedback to students for design-driven problem formats. These design-driven learning examples are prevalent in Electric Circuits (ENGR:21229), Software Design (ECE:3330), Computers in Engineering (ENGR:2370), and Introduction to Engineering Computing (ENGR:1300) are needed.

Core PrairieLearn customizations are needed to tailor the environment to the needs of the University of Iowa. The primary developer of the PrairieLearn environment, Matt West, has provided support for integrating these customizations into the main package.

Training materials for accelerating faculty adoption of this tool into the courses will be generated. Training materials for helping student best use this environment will be created (i.e. training videos for configuring a VPN that will allow off-site remote learners to access homework and test questions).

This project deploys a testing infrastructure that facilitates a partial return to free response questions by providing real-time feedback to students, auto-grading or assisted grading. The PrairieLearn environment allows for complex question programming needed in STEM-related fields where complicated mathematical answer or programming free response questions can be asked and automatically or semi-automatically graded.

A recent Online Testing Committee chaired by Annette Beck at the Office of Teaching, Learning and Technology have not identified a vendor product which adequately meets the electronic testing needs of the College of Engineering.

How will it improve student learning?:

The PrairieLearn platform provides an interactive problem-solving environment for STEM-based problem sets. This setting is available for homework, test-preparation, or administering tests. The PrairieLearn platform allows the examination experience to also act as a teaching opportunity by providing real-time feedback concerning potential answers.

The proposed "speed-grader" provides the necessary efficiencies to allow faculty the opportunity to provide individualized feedback on a per problem basis rapidly for large student numbers.

Each homework question can have multiple numbers (potentially infinite) of variations. Under this evaluation approach with instant auto-grading, students can be prompted with a new adaptation of a question until they answer it correctly. Learning is improved by allowing students to retry problems until they master the underlying concept or method. Once a student achieves proficiency, they should be able to answer similar questions very quickly. Within PrairieLearn, each problem has a value, a point total, and a point maximum. If you answer a question correctly, two things happen; The total point value increases until you reach the point maximum. The value increases (2x, 3x, 4x). If you answer a question incorrectly, one thing happens: The value goes back to what it was initially (1x). This approach emphasizes mastery of topics over memorization of exemplary cases. The PraireLearn system rewards repeated correct answers, which tend to demonstrate ability. There is no penalty (other than resetting the value) for answering a question incorrectly, and students receive no penalties for submitting multiple incorrect answers. Rewards are achieved through correct solutions.

Once a design-driven problem is developed to provide programmatically defined solutions, it is simple to generate an infinite number of unique questions from that template. This feature allows the use of a problem in both self-paced homework and time-limited examination environments.

Using PrairieLearn for test evaluation also provides instant feedback to students so that the testing experience is also a learning experience. Multiple-choice questions require artificially constructed problems that must have single solutions. The ability to build open-ended design problems allows students to develop and display their mastery of the engineering topics in a way that is more natural to the underlying skills needed.

What resources will you need?:

Student developers, likely from the College of Engineering or the Computer Science department, will be required to implement the interfaces and improvements to PrarieLearn as well as assisting faculty with generating banks of questions for specific exams.

Core IT infrastructure of virtual machine hosting and systems support will be needed from IT staff within the College of Engineering. The Chief Technology Officer of the College of Engineering has committed these resources at no additional cost to the project.

What is your rough estimate of costs?:

$1143.5 – cost of travel and room/board/Per-Diem for two individuals visit the University of Illinois for 2 days, to consult with their faculty and staff regarding the electronic testing service.

$5193.60 – 16 Weeks of 2 10hr/week RA’s to develop templates for each of the ENGR:2120 Electronic Circuits course homework and exam topics.

$2596.8 – 16 Weeks of 1 10hr/week RA’s to develop templates for each of the ENGR:2730 Computers in Engineering course topics and exam topics.

$1298.40 – 16 Weeks of 5hr/week RA’s to develop templates for each of the ECE:3330 Software Design course topics and exam topics. (This is much less due to overlap with ENGR:2730)

$1298.40 – 16 Weeks of 5hr/week RA’s to develop templates for each of the ENGR:1300 Engineering Computing course topics and exam topics. (This is much less due to overlap with ENGR:2730)

$5193.60 – 16 Weeks of 2 10hr/week RA’s to adapt the Perseus like question editor/rendering for facilitating exam question generation.

$1298.40 – 16 Weeks of 5hr/week RA’s to develop training materials for both faculty and students to improve adoption and utilization of the platform. The learning curve for developing high quality engaging problems is a risk to this project that will be addressed through extensive training materials. These students will also provide administrative support for organizing the project.

TOTAL: $24514.70