

An Agile/Scrum Method of Instructional Design for Multimedia Learning via Computer Based

Learning

Kierstin Bible

University of Arkansas

ETEC 5243 - Instructional Design Theory & Models

October 8, 2017

An Agile/Scrum Method of Instructional Design for Multimedia Learning via Computer Based Learning

Richard Mayer's research on multimedia learning began with a simple thesis statement: "People learn better from words and pictures than from words alone" (Mayer, 2009, Preface). His research led to a cognitive theory of multimedia learning which pulls together dual coding theory, cognitive load theory and constructivist learning theory and ultimately resulted in the creation of Mayer's Twelve Principles of Multimedia Design. Multimedia design lends itself extremely well to computer based learning (CBL), but CBL can be difficult to create in a traditional design model, where the basic design does not change once development begins (Brown & Green, 2016, p. 8). In the case of designing multimedia CBL courseware for university classes, the design model must have flexibility built in, with the opportunity for frequent formative evaluation that allows the development team to make changes as needed for maximum learning outcome (Brown & Green, 2016, p. 10)

In the world of project management, this kind of adaptive method has been aptly named "Agile" with emphasis on "early and continuous delivery of valuable software", "welcom[ing] changing requirements, even late in development", simplicity, and "self-reflection on how to make the team and development process more effective" (Pham & Pham, 2012, p. 4-5). More specifically, the use of Scrum within Agile project management focuses on short development cycles, the result of which are "potentially shippable products" (Pham & Pham, 2012, p. 8).

The idea of two to four week development iterations (or "sprints" as they are called in Scrum) lends itself extremely well to Mayer's emphasis on cognitive load theory or the idea that "working memory can process only a few elements at any one time" (Mayer & Moreno, 2002, p. 116). Concentrating on ten to fifteen minute learning modules that follow Mayer's principles

will allow fairly simple modules to first be developed and then tested for usability. Mayer's emphasis on the lack of extraneous images and sounds (the coherence principle) (Mayer, 2009, Chapter 4) (Mayer & Moreno, 2002, p. 114), as well as the avoidance of on-screen text accompanying graphics and narration (the redundancy principle) (Mayer, 2009, Chapter 6) (Mayer & Moreno, 2002, p. 115) will help the development team to avoid unnecessary elements and keep the process on track.

Scrum-based design models have been tested in the world of courseware and instructional design with positive results, such as the AM-OER model, which focused on developing open educational resources (Arimoto, Barroca, & Barbosa, 2016, p. 205), and work in the Federal University of Lavras, which focused on the production of learning objects for blended learning (de Bettio, Pereira, Martins, & Heimfarth, 2013, p. 38). Building multimedia CBL for a university course (graduate or undergraduate) in an environment that has the essential resources of instructional designer(s), project manager(s) and a development team that includes a media/video team, animation specialists, graphic designers and software developers, can be accomplished using a Scrum-based instructional design model that allows for continuous feedback from the instructional designer to the project manager and development team along with formative evaluation of each module.

Analysis

Analysis: Needs, Task and Learner Analysis

The foundation of this process is the analysis phase, beginning with the needs analysis to determine whether CBL is the best option to deliver the most effective multimedia learning experience. The instructional designer will perform this analysis in tandem with the task analysis, looking at the instructional goals and objectives set out by the instructor and

determining whether multimedia CBL is needed for some or all of the course content. It is at this point that the designer first thinks about the overall architecture of the CBL modules. Also under consideration is the learner analysis, although this is partially already accomplished simply by the intended students for the course. Are they graduate or undergraduate? Lower level or upper level? Is this core curriculum? A course for a specific major? Questions like these will also affect the overall architecture of the modules as well as the content of each individual module.

Design

Architecture Creation, Approval & Initial Storyboards

After meeting with the instructor, establishing the course goals and objectives, and performing the learner analysis, the designer will begin the process of laying out the basic architecture of the modules. S/he will determine the overall number of modules (based on course objectives) and lay out the basic structure for order of modules, design ideas and length.

Once the basic design is achieved, the designer will arrange another meeting with the instructor to begin the approval process. This is the point where the designer must thoroughly explain the Mayer's principles as they apply to each module, helping the instructor to understand and ultimately approve the design choices for each module. There may ultimately be several meetings with the instructor, each looping back around to the module architecture and storyboard until the "final" version is approved.

Backlog Creation

After the overall design is approved, the designer will sketch out the storyboards for each module, tying each to one or more course objectives. S/he will create a very basic module backlog based upon his/her understanding of the needs of each one. Project management software that allows one to easily create, duplicate and move elements would be especially

helpful in this process. This initial backlog can be shared with the project manager before the first meeting with the designer, and may also be viewed by the design team to give them an idea of what's in the pipeline.

Overview Meeting with Project Manager

After the “final” version of the course is agreed upon by the instructor and designer, and the backlog has been started, the designer then meets with the project manager to refine the backlog, based upon the project manager's knowledge of the development process and what may be required. The project manager will establish the makeup of the development team which may include a media team, software developers, animation specialists and graphic designers. The decision is made at this time how many of each team will be needed and who will lead each one.

It is at this time that the project manager will establish the overall scope, budget and timeline for the project, based upon the information provided by the designer and the decisions made concerning the development team. If the planned project outstrips the resources of time, budget and personnel, the designer will need to consult with the instructor, make changes and once again meet with the project manager. However, if the courses are set up with balanced content and objectives (i.e. not trying to deliver too much content in one semester), this should be easy to avoid. A course that cannot be paired down to the required budget, time and personnel may need to be split up over one or more courses. This process should be laid out according to the processes established by the university and the state in which it resides and will not be included within this model.

After the final meeting in the design phase between the designer and project manager, the backlog that was initially created has been filled in by the project manager and is ready for the Project Kickoff Meeting.

Development, Implementation & Formative Evaluation Cycle

Project Kickoff Meeting

After the budget, scope and timeline have been set, with the basic module architecture planned (including storyboards for each), the project kickoff meeting begins the development process. It is at this meeting that the entire development team is introduced to the designer, possibly for the first time, depending on personnel turnover and the size of the university. The designer will cover the course objectives and overall module architecture with the team, showing them the “big picture” and then going into detail on the first module, including storyboards. The development team will help further refine the backlog and establish the time-box for the first Module Sprint, which should be between two and four weeks.

Module (1) Sprint

One of the ubiquitous feature of Scrum is the idea of a “Daily Standup” meeting, in which the project manager meets with the team to discuss the project (Pham & Pham, 2012, p. 8). In this design process, the daily standups are included, with representatives from each team (possibly team leaders, depending upon the direction/preference of the project manager) contributing to this short 15 to 30 minute meeting. Each team member is asked to answer three basic questions: What did you do yesterday? What are you planning on doing today? What impediments are in your way? In this way, the project manager can help keep the project moving, clearing “blockers” as needed for the team.

In addition to the daily standup, there may be meetings as needed between members of the development team, the project manager, the instructional designer and the instructor. These will likely flow out of the efforts of the project manager to remove blockers for the development team.

Module (1) Testing Level 1 (Formative Evaluation)

After the Module Sprint has concluded, the module needs to go through several levels of formative evaluation with the development team, the instructional designer and the instructor. The development team needs to evaluate it on the most basic standards – does it work as intended? This evaluation should actually performed as each small piece is programmed, and then again after the module is completed. Once it is determined that it performs as intended, the designer will test the module, ensuring that it follows the design storyboard. Finally, the instructor will have the chance to work through the module. Each of these evaluations may result in edits being requested, with additions being made to the backlog. Only at the point that the instructor and designer agree that the module performs as intended, meeting its course objective(s), does the module go into usability testing with students.

Module (1) Usability Testing (Informal implementation & formative evaluation)

After each module goes through the circular development and evaluation process, it goes into usability testing with its intended learners, the students of the program. Volunteers should be relatively easy to acquire with the promise of extra credit – or the testing may be worked into the curriculum of a different course.

Ideally each testing session is recorded (i.e. with software such as Camtasia, which will record what is on the monitor as well as the speech of the user). Students will be encouraged to “talk through” any issues as they encounter them. The instructional designer will view the recordings, finding spots where students have had issues. He may conduct student interviews to clarify what was misunderstood or difficult. These recordings will accompany student surveys, which the designer will also review. After the testing series has completed for the module in

question, the designer will need to determine if any further edits need to be made (which will go into the backlog) or if the module can be counted as “complete.”

Module (1) Sprint Retrospective

An especially important part of the Scrum process that should not be left out of this design process is the sprint retrospective, which is a meeting in which the development comes together to address issues with communication, time-box estimates, design confusion, etc. If this meeting is handled well by the project manager and if the team feels “safe,” many issues can be overcome leading to a smoother and quicker process for remaining modules and for the function of the development team in general.

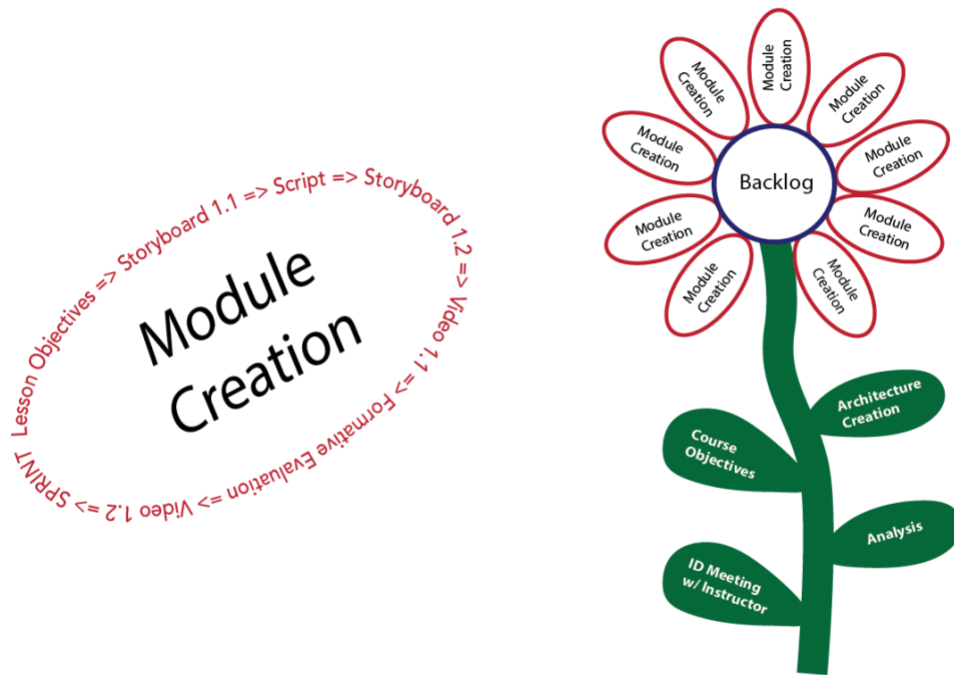
Backlog Grooming

After the previous module has started the evaluation phases, the backlog for the next module can be “groomed” in preparation for the next kickoff meeting by the project manager. This may be based on experience in the development process and/or further meetings with the designer.

Module (2) Kickoff

After the backlog has been groomed for the next module, the development team, project manager and instructional designer “start” the process over, with a kickoff meeting that covers the storyboard of the new module, refines the backlog and sets the time-box for the upcoming module sprint. More than one module can be in the development process, as one is tested/edited and one is initially developed. It is up to project manager to ensure the development team is not overwhelmed and can keep to the time-box deadlines. Meanwhile, the instructional designer can be at the analysis, design or evaluation phase of other CBL projects. Each module circles

through the development and evaluation process, coming back around to the overall backlog until the complete module architecture is complete and the CBL has taken its final shape:



Formal Implementation and Summative Evaluation

Once the CBL has been officially launched (i.e. delivered to students in the university in the course to which it belongs), it enters the summative effective in creating successful multimedia learning. A combination of student surveys (i.e. end-of-semester evaluations), test/paper scores and instructor feedback are collected by the instructional designer to determine if any further edits are needed. One or several modules may go back to the development team for changes, becoming a part of the cycle once again.

Conclusion

This paper describes a version of Scrum project management that has been specifically designed for Computer Based Learning in university courses, both graduate and undergraduate. Instead of performing both formative and summative evaluation after the complete CBL has been developed, evaluation is performed throughout the development process, allowing for changes to make the final product an effective learning experience. It overcomes the shortcomings of the a linear model and allows the building of effective learning modules in a relatively short period of time (Durdu, Yalabik, & Cagiltay, 2009, p. 230), which ensures that the final product will be both what the instructor intended, will actually work and will be effective.

The entire process takes a skilled project manager to monitor what may be several courses being developed at different points in their process – initial module creation, edits of modules, updates/edits of previous courses/modules, etc.

References

- Arimoto, M. M., Barroca, L., & Barbosa, E. F. (2016, March). AM-OER: An agile method for the development of open educational resources. *Informatics in Education, 15*(2), 205-233. <http://dx.doi.org/10.15388/infedu.2016.11>
- Brown, A. H., & Green, T. D. (2016). *The essentials of instructional design: Connecting fundamental principles with process and practice* (3rd ed.). New York: Routledge.
- de Bettio, R. W., Pereira, D. A., Martins, R. X., & Heimfarth, T. (2013). The experience of using the scrum process in the production of learning objects for blended learning. *Informatics in Education, 12*(1), 29-41. Retrieved from <http://0-search.proquest.com.library.uark.edu/docview/1352758723?accountid=8361>
- Durdu, P. O., Yalabik, N., & Cagiltay, K. (2009). A distributed online curriculum and courseware development model. *Journal of Educational Technology & Society, 12*(1), 230-248. Retrieved from <http://0-search.proquest.com.library.uark.edu/docview/1287038880?accountid=8361>
- Mayer, R. E. (2009). *Multimedia learning* (2nd ed.). [Kindle]. Retrieved from <http://www.amazon.com>
- Mayer, R. E., & Moreno, R. (2002, February). Aids to computer-based multimedia learning. *Learning and Instruction, 12*(1), 107-119. [http://dx.doi.org/10.1016/S0959-4752\(01\)00018-4](http://dx.doi.org/10.1016/S0959-4752(01)00018-4)
- Pham, A., & Pham, P. (2012). *Scrum in action: Agile software project management and development*. [Kindle DX]. Retrieved from Ebscohost