VALA2018 LIBRARIES/ TECHNOLOGY AND THE FUTURE// 13-15 FEBRUARY MELBOURNE CONVENTION & EXHIBITION CENTRE

Iterative and incremental evaluation works for software development, but can it be good for student learning initiatives in Australian academic libraries?

Kathryn Greenhill
Lecturer, Information Studies
Curtin University
k.greenhill@curtin.edu.au
http://orcid.org/0000-0001-9357-6006

Karen Miller
Curtin University Library
Curtin University
Karen.Miller@curtin.edu.au
http://orcid.org/0000-0002-2094-2455

Abstract:
Iterative and incremental development in software engineering involves small, ongoing "evaluate, review, act" cycles, allowing rapid development of rough prototypes of a software product that can be altered and re-tested, long before the product is considered "finished" and made available to the final stakeholders. This paper investigates whether Australian academic libraries are currently applying iterative and incremental evaluation to the development of student learning initiatives run by the library. It examines whether there are possible places in the development-cycle of these initiatives where iterative evaluation could happen, and whether it actually does happen.

PEER-REVIEWED

First published 1 February 2018

Introduction

Iterative and incremental evaluation is a key part of many software development methodologies, particularly those methods that can be grouped together as “agile” (for example Crystal Clear (Cockburn, 2004), Scrum (Schwaber and Beedle, 2001), and Lean (Poppendieck and Poppendieck, 2013)). Agile methodologies aim to work in ways that could be advantageous to many projects and organisations in other arenas, with a focus on flexibility, achieving working outcomes swiftly, daily meetings, face-to-face communication and involving all stakeholders in all stages of the process (Beck et al., 2001). While there have been suggestions that agile methodologies in general could be applied to library work (Cervone, 2011, 2014; Chang, 2010), the current investigation focuses chiefly on just one aspect of agile, the timing of evaluation, specifically as this relates to the concepts of Iterative and Incremental Development (IID).

Whether flexible, iterative timing in a methodology used in a different field (computer programming) to achieve different aims (output of working code) can be usefully adapted to librarianship and library projects is an interesting question. This research aims to provide a small component toward answering this, by determining whether Australian academic libraries already exhibit this kind of flexible timing in their projects aimed at student learning.

We wanted to find out whether project management in library services uses a more rigid, waterfall-based model that typified early software development projects. This would be a model where all goals are set before development commences, then only when work on one milestone is completely finished would work on the next commence, without any review of the scope or aims of the project in light of what has been learned in the achievement of the first goal.

Background

Iterative vs incremental development

The major strengths of iterative development and incremental development are speed, responsiveness and flexibility (Osorio et al., 2011). These two concepts are closely related and often discussed as part of the same process, hence the acronym IID (Iterative and Incremental Development).

Incremental development involves breaking a project into very small, achievable parts that may affect the later development path of the project. Although it is clear just what will be developed in the next “sprint”, the remaining steps are left somewhat vague so that any discoveries or improvements emerging in the small step can be integrated into the software easily, and the improved path implemented nimbly. By contrast, the similar concept of *iterative development* involves the constant review of functionality and quality of what is being produced. It involves returning at each step along the way to earlier stages, to test the effectiveness of the plan and output in the light of how the project is evolving, and making changes in response to these observations. So, incremental development is more about timing

of project effort (in very small chunks), while iterative development is more about what is done in response to these small chunks, recalibrating often, regularly and in response to new knowledge at the time it is gathered.

The concepts of *iterative evaluation* and *incremental evaluation* can be similarly differentiated. Incremental evaluation refers to checking what is being achieved at regular, short intervals. Iterative evaluation, by contrast, involves acting upon the information obtained in an incremental evaluation to work out whether any changes could improve the project.

Historical "single-pass waterfall" software development

In exploring whether libraries are evaluating iteratively and/or incrementally, it is useful to consider the presumptions and conditions that led to IID being applied so effectively in software development.

In the late 1960s, while "hackers" were part of a coding community around universities that focused on experimentation in a rather anarchic community that valued sharing (Levy, 1981), there was another, more sober, stream of software development. This involved bespoke programs for government and sometimes business, often executed by more formally organised private companies or organisational units within government departments (Larman and Basili, 2003). Inherited military project-management methods meant complex problems needing complex solutions were stymied by simplistic timelines and presumptions about software development (Brooks, 1995, p. 14).

In these circumstances, software was often developed to specifications in invitations-to-tender that were written by non-programmers. These documents, controlling the project outcomes, made presumptions about developmental processes that did not match realities of software development (Wong, 1984). Testing, and comparison to the client specifications, was a stage only at the very end of the project. Larman and Basili (2003, p. 47) characterise this as "a single-pass sequential, document-driven, gated-step" approach.

This "single-pass waterfall" style of development did not suit very complex projects where best solutions, or even product requirements, could often not really be known until field-testing. As Wong (1984, p.723) explains, "in reality software development is a complex, continuous, iterative and repetitive process".

Iterative and incremental software development and "agile" methodology

Iterative software development models have been around since the 1940s, when several developers separately explored the application of Walter Shewhart's "plan-do-study-act" cyclical model that had been successfully used in the Bell Labs in the 1930s (Larman and Basili, 2003). Although IID is a flexible alternative to the waterfall model, discussions of alternatives tend to focus on a group of semi-formal and lightweight methodologies collectively described as "agile". These include Crystal Clear (Cockburn, 2004), Scrum (Schwaber and Beedle, 2001), and Lean (Poppendieck and Poppendieck, 2013). These methodologies all use IID, but also

have other common stated values, such as a focus on people rather than on process.

Published in 2003, the Agile Manifesto (http://agilemanifesto.org/) was created by a number of developers over a weekend and consists of a summary of four values and twelve principles. The four values are:

- *Individuals and interactions over processes and tools*
- *Working software over comprehensive documentation*
- *Customer collaboration over contract negotiation*
- *Responding to change over following a plan*

Application of agile methodologies outside of software development

Although historically used to develop software, the advantages of rapid-review, communication-based iterative methods of working are not inherent to software development. In an interview with four of the key authors of the Agile Manifesto ten years after its publication, all agreed that agile has an application to projects beyond software development. Jim Highsmith identified the type of project that benefits best from agile "practices and principles" as "any project that faces uncertainty, complexity, volatility and risk" (Jackson, 2012, p. 61) Andrew Hunt emphasised that "[a]t its heart, an agile approach has little to do with software; it's all about recognizing and applying feedback.... Agile development uses feedback to make constant adjustments in a highly collaborative environment" (2012, p. 62).

Agile methodology is being considered in some arenas outside of software development, such as teaching mechatronics (Edin Grimheden, 2013), commercial aircraft product development (Carlson and Turner, 2013) and health interventions using mobile devices (Jacobs and Graham, 2016). It has attracted the interest of learning designers as a more flexible alternative to the long-standing ADDIE (Analysis, Design, Development, Implementation, Evaluation) model. Yocum (2015) explored how agile principles, processes and practices are used by some instructional design teams to promote flexibility and creativity in developing instructional design projects, while Arimoto et al (2016) have studied its use for the development of Open Educational Resources.

Within the library sphere, there has been some use of iterative evaluation or agile methodology outside of developing software. Cervone (2014) outlines a specific agile methodology (Scrum) and made theoretical suggestions about how this could be applied to library strategic planning. Chang (2010) describes how a small team in an academic library successfully implemented disparate technological projects (e.g. A Wiimote interactive whiteboard, Helpdesk call log, Learning spaces website) using a specific agile method, Crystal Clear. She concludes that "this lightweight and adaptive software development and project management approach enabled us to undertake rapid development and deployment of quality applications with a small team" (p.688). Researchers have also used iterative evaluation approaches to design, develop and evaluate digital libraries (Carlo Bertot et al., 2006; Somerville and Brar, 2009) as well as library websites (Gallant and Wright, 2014; Manzari and Trinidad-Christensen, 2006).

Evaluation methods and timing for library programmes

One of the most common reasons given for evaluating library learning programmes, alongside determining whether programme outcomes or goals were met, is to provide strategies for improvement (Bober et al., 1995; Oakleaf and Hinchliffe, 2008). However, the majority of case study literature that describes evaluation of library programmes "do not always provide detailed information about the last part of the assessment cycle – using the results to make improvements" (Swoger, 2011, p. 245). Nonetheless, an iterative approach, to which change is inherent, is considered best practice when evaluating library learning programmes.

In theory, iterative evaluation allows librarians to "get started" with assessment rather than waiting to "get it perfect", while "[e]ach repetition of the assessment cycle allows librarians to adjust learning goals and outcomes, vary instructional strategies, experiment with different assessment methods, and improve over time" (Oakleaf and Kaske, 2009, p. 286). Oakleaf's ILIAC (Information Literacy Instruction Assessment Cycle) model is based on these principles, and includes seven stages that loop back to the beginning, where the cycle begins anew (Oakleaf, 2009).

Gustavson (2011) applied the ILIAC model to the creation of an online information literacy multi-media tutorial, by conducting a pre-test in the design phase that enabled instructors to identify student deficits, address these in the content, and make revisions for the next iteration. Ondrusek (2005) describes developing a quiz as an iterative process requiring "systematic evaluation and subsequent revision of individual quiz items and the accompanying instruction" (p. 391). Further case studies illustrate iterative approaches giving rise to changes in programme goals and learning outcomes, content, and delivery (Lei Hsieh and Holden, 2010; Salisbury and Ellis, 2003; Swoger, 2011).

In most case studies, however, the changes mainly occurred at the end of the delivery for the next iteration or delivery of the program, rather than during the process itself. As Swoger (2011) points out, the changes "mark the final part of the assessment cycle" as part of "closing the loop" (2011, p. 245). Arguably, this common application of iterative evaluation is not fully-fledged IID development.

The new Framework for Information Literacy from the Association of College and Research Libraries (ACRL) (2016) represents a significant shift in thinking about evaluation that provides for a genuinely iterative approach. It shifts focus away from student achievement of a fixed set of competencies, skills, and outcomes toward "threshold concepts":

> *Consideration of threshold concepts to some extent 'rattles the cage' of a linear approach to curriculum design that assumes standard and homogenised outcomes… We would argue … for the notion of learning as excursive, as a journey or excursion which will have intended direction and outcome but will also acknowledge (and indeed desire) that there will be deviation and unexpected outcome within the excursion; there will be digression and revising (recursion) and possible further points of departure and revised direction.*
>
> (Land, Cousin, Meyer and Davis, 2006, p. 202, cited by Oakleaf, 2014, p. 511).

While Oakleaf provides the ILIAC model in relation to assessment, Char Booth's model of library learning programme design, the USER model, is also useful in guiding an iterative approach to the learning design process (Booth, 2011). The four overlapping stages of Booth's USER instructional planning cycle are:

1. Understand (investigate learner needs and the learning scenario);
2. Structure (specify content and create support strategies);
3. Engage (design and deliver the content); and
4. Reflect (investigate evidence of learning, assess learning outcomes, consider how to revise, improve and reuse content and materials).

The evaluation of learning programs in Booth's model is most focussed in the final 'Reflect' phase, however different evaluation methods "are deployed at different points along the instructional cycle" forming a "feedback loop" (p. 138). In order to better understand and respond to learners, the feedback loop should be a process of iterative development, "building consistent and multiple opportunities to gauge comprehension and engagement based on pre-defined targets as well as in-the-moment observation, then revising or supplementing your approach if necessary" (p.139). The model describes possible opportunities for iterative evaluation at four more-or-less sequential phases: "Pre-assessment", "Formative", "Summative" and "Confirmative".

Like Oakleaf and Kaske (2009), Booth advocates using multiple evaluation methods at different times through the process, arguing that it "gives better insight into the learner experience" (p. 149). Different evaluation methodologies can be used to evaluate "Learner" (the recipients of the instruction), "Context" (the environment of instruction), "Content" (the skills, knowledge and attitudes the learner will gain) and "Educator" (resources and support from colleagues). Focussing on how, when and what is evaluated in student learning initiatives in libraries, Booth (2011) offers a useful model upon which to build an inquiry into these aspects, and whether changes were made as a result. This framework was adapted for our survey, as outlined in the methodology section below.

Methodology

Studies looking at evaluation methods in libraries have used a variety of methodologies including literature reviews (Bober et al., 1995; Hufford, 2013), survey methodology using interviews (Savage et al., 2017) and questionnaires (Farkas et al., 2015; Oakleaf and Hinchliffe, 2008). A survey methodology is an appropriate method to ascertain evaluation practices, as it is a method that is "used to determine the present status of a given phenomenon and can make inferences about a larger group from a smaller sample group (Connaway et al., 2016, p. 97).

This study used an online survey to investigate the self-reported timing and application of evaluation in student-learning initiatives delivered by Australian academic libraries.

Participants

All 39 University Librarians listed on the Council of Australian University Librarians (CAUL) members directory pages (http://www.caul.edu.au/about-caul/caullist), which constitutes "all University Librarians or equivalent of institutions which have representation on Universities Australia" (Council of Australian University Librarians, 2014), were contacted by email to request that they nominate a staff member to complete an online survey.

Type of student learning initiative

We asked that candidates nominated were involved, if possible, in the specific type of learning initiative that we were examining. This activity would need to include all of the following:

a) student engagement in a learning activity (students had to actually do something as part of the programme, rather than sit in an audience as passive listeners).
b) introduction of substantially different content or delivery methods when compared to other projects or programmes within the five years before the initiative was run
c) was delivered between 1 July 2010 and 30 June 2015

This type of initiative was chosen for five main reasons:

1. Incorporating student learning activities gave a possible point where the learner could be evaluated, and this information fed back into the way the rest of the programme was delivered. There was thus a clear opportunity to check whether iterative evaluation of the learner was actually happening when there was opportunity
2. An initiative like this is likely to have some degree of complexity and unpredictability, with the actual behaviour and reception from the students likely to be an unknown at the initial planning stage, replicating on a small scale some of the uncertainty and complexity that makes iterative and incremental development a sensible design choice for software
3. These kinds of projects are more likely to be recently implemented, given that libraries are slowly changing the focus of their information literacy activities to include greater student participation (Kaplowitz and Kaplowitz, 2012, p. 11). Recency meant it was more likely that library staff involved in the initial implementation were still available to provide data
4. This also ruled out projects that had been conducted with little change for a number of years. We reasoned that more recent projects would give opportunity for the designers of the initiatives to have chosen newer methods of project-design and evaluation if they wished to.

Survey content

The survey consisted of three parts. The first part asked about the initiative itself (for example objective and target population). The third part used mainly free-text and multiple choice questions to determine participants' opinions of the usefulness and outcomes of their evaluation of the learning initiative. This paper is concerned chiefly with the second part of the survey, described below.

Matrices of time of evaluation by type of evaluation for four aspects

In the second part of the survey, participants indicated on four matrices (one for each of four aspects examined) the time point at which they evaluated a particular aspect (if at all) using each of fourteen possible methods presented.

We used the framework in Booth's (2011) USER model to structure these matrices, slightly changing the names of Booth's four broad aspects to be evaluated (Learner, Context, Content, Educator) and describing them to participants in this way:

- the LEARNER (e.g. existing knowledge and skills, preferred content, preferred method, competencies achieved);
- the METHOD OF DELIVERY (e.g. timing, location, length of session);
- the CONTENT (e.g. relevance, depth, complexity, missing topics);
- the LIBRARY STAFF INVOLVED (e.g. communication skills, attitude, preparedness)

As discussed earlier, Booth (2011) had identified a range of possible methods of evaluation. We summarised and condensed these into the fourteen methods shown in Figure 1 (the matrix for the aspect, LEARNER) below.

We used four time periods, as did Booth. The time periods in the survey differed from Booth's more conceptual distinction (based on aims and techniques) in that they were differentiated purely by chronology, giving us *'before', 'during', 'immediately after,* and *'more than 24 hours afterward*'.

In each matrix, participants used check boxes to indicate the time point for each evaluation method in each of the four matrices for each aspect. So, for example, the participant could indicate that the LEARNER was evaluated by using an "activity evaluating participant skill" before, during and more than 24 hours after the programme.

Q10.
Please select all methods you used to evaluate the LEARNER. For example, existing knowledge and skills, preferred content, preferred method.

	before	during	immediately after	more than 24 hours after	not used	not applicable
Survey of potential participants	☐	☐	☐	☐	☐	☐
Survey of actual participants	☐	☐	☐	☐	☐	☐
Conversation with actual participant(s)	☐	☐	☐	☐	☐	☐
Conversation with academic staff	☐	☐	☐	☐	☐	☐
Reflection by library staff member responsible for initiative delivery	☐	☐	☐	☐	☐	☐
Formal observation and feedback by other library staff	☐	☐	☐	☐	☐	☐
Informal observation and feedback by other library staff	☐	☐	☐	☐	☐	☐
Small-scale pilot version of planned activity	☐	☐	☐	☐	☐	☐
Research into other similar projects (eg. web searching, literature review, informal discussion)	☐	☐	☐	☐	☐	☐
Discussion with an external community of practice	☐	☐	☐	☐	☐	☐
Computer generated analytics (eg. automatic count of site hits, number of logins, percentage completion rate)	☐	☐	☐	☐	☐	☐
Activity evaluating participant skill	☐	☐	☐	☐	☐	☐
Social media backchannel	☐	☐	☐	☐	☐	☐
Other (specify)	☐	☐	☐	☐	☐	☐

Figure 1. The survey matrix for the "LEARNER" aspect of the student learning initiative.

The selection options in each row (evaluation method) and column (time period) were identical for each of the four aspects. This made responding to the survey easier and swifter for participants, as respondents needed only to familiarise themselves with the wording of the first matrix to be able to respond to all four. However, not all methods of evaluation presented were possible for all time periods, or as easy, for every aspect. (For example, in the figure above, evaluating a LEARNER *during* the initiative delivery via the method "discussion with an external community of practice" would be neither professional nor really possible).

In the results, a weighting that allowed for whether a particular method was impossible or very difficult was applied to the raw totals of evaluation incidents recorded. This provided clearer comparison of evaluation at each point that was due to design decision.

Results

Response rate

A total of 39 Australian academic libraries were contacted, constituting all major university libraries in the country. Twenty out of 39 institutions responded. After allowing for duplicates, unfinished attempts and institutions without suitable initiatives, this left a response rate of 14 institutions (36%). This response rate is aligned with the benchmark suggested by Baruch and Holtem (1999; 2008), whose

analysis of survey response rates as reported in a range of literature across two disciplines found that survey responses when sent to organisational representatives are typically lower (36% average) than when sent to individuals, where the average response rate was 55%.

When desired outcomes were decided

Participants were asked the point at which they decided the desired outcomes of the initiative. Results are summarised in Table 1 below.

Q9 - When did you decide the desired outcomes of the initiative?	
As part of the initial initiative brief	9
During development of the initiative	4
Just before the initiative was delivered	0
During delivery of the initiative	0
After the initiative was delivered	0
It was not considered at any point	1
Total	**14**

Table 1. Time at which desired outcomes were decided

Raw totals of incidents of evaluation by time point and aspect

A total of 514 incidents of evaluation happened across the fourteen learning initiatives. This is an average of 36.7 evaluations for each library by any method of any aspect at any time point.

Table 2 shows the collated incidence of evaluation by any method for the different time points for the different aspects examined. This paper does not further discuss separate results for each of the fourteen different options for evaluation methods, as the distinction does not need to be made to discuss the timing at which any evaluation happened, but will be covered in further work from these results.

		TIMING OF EVALUATION				
		before	during	within 24 hours	more than 24 hours after	total used any time
ASPECT EVALUATED	Learner	49	30	25	43	147
	Content	54	19	26	48	147
	Method	52	19	30	44	145
	Library Staff	27	18	12	18	75
	TOTAL	**182**	**86**	**93**	**153**	**514**

Table 2: Raw total of incidences of evaluation recorded for each aspect by time point

Weighted comparative incidents of evaluation by time point, by aspect

To validly compare incidences of evaluation between the aspects, we applied a weighting to the raw figures to allow for the fact that some methods of evaluation were more difficult (or even impossible) for some aspects or time points. As discussed in the methodology, not all aspects could equally be evaluated by each method at each time (For example, LEARNER could be evaluated by "activity testing participant skill" *during* the initiative, but LIBRARY STAFF could not).

The adjusted raw scores are below, rounded to two decimal places. It is vital to remember that the figures are no longer actual reported individual instances of an evaluation happening, but normalised comparison figures that better indicate differences in frequency and timing of evaluations due to library staff design decisions. The method for obtaining and applying the adjustment factor is detailed in Appendix One *(Calculation of weighting of reported incidents of evaluation by time point by aspect.)*

		TIMING OF EVALUATION				
		before	during	within 24 hours	more than 24 hours after	total used any time
ASPECT EVALUATED	Learner	57.17	46.67	29.17	50.17	183.17
	Content	58.15	26.60	30.33	56.00	171.09
	Method	56.00	26.60	35.00	51.33	168.93
	Library Staff	31.50	36.00	18.67	28.00	114.17
	TOTAL	**202.82**	**135.87**	**113.17**	**185.50**	**637.35**

Table 3: Weighted total numbers of any incidence of evaluation recorded for each aspect for each time point

As can be seen from the pie chart in Figure 2, evaluation did happen during each time period considered, with the most happening before the initiative was delivered (32%), and the least happening within 24 hours of the initiative. Combining figures for the two time periods that occurred after the initiative was run results in a combined weighted incidence of 47% of all evaluations happening after the initiative was run.

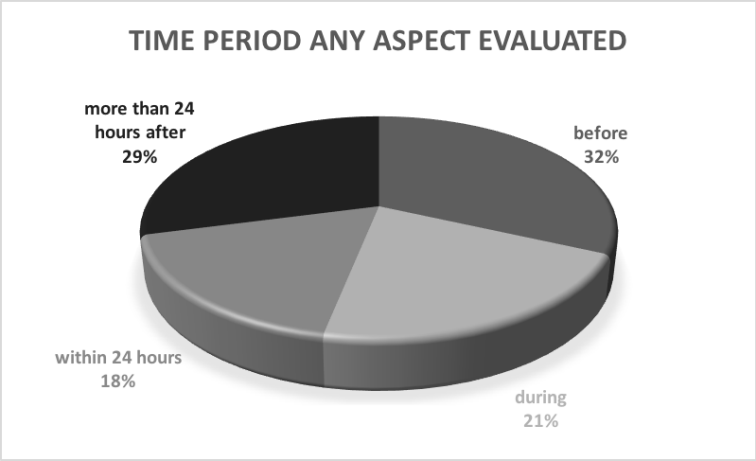

Figure 2: Time period any aspect evaluated (weighted totals)

Changes to method of delivery and content

When participants were asked whether they changed aspects of the learning initiative in response to evaluation, the majority responded that they did not. Specifically, when asked about changing the planned method of delivery, 57.14% said they did not make any changes, while when asked about the content of the material, over 50% said they did not make any changes (see Tables 3 and 4 below).

Q20 - Did you change the planned METHOD OF DELIVERY in response to evaluation?		
Response	**Percentage**	**Number**
Yes, before delivery	7.14%	1
Yes, during delivery	0.00%	0
Yes, after delivery	21.43%	3
No, but would change it if run again	14.29%	2
No, and would not change it if run again	57.14%	8
Total	**100%**	**14**

Table 3: Timing of any changes to METHOD OF DELIVERY

Q22 - Did you change the planned CONTENT OF THE MATERIAL in response to evaluation?		
Response	**Percentage**	**Number**
Yes, before delivery	0.00%	0
Yes, during delivery	0.00%	0
Yes, after delivery	42.86%	6
No, but would change it if run again	7.14%	1
No, and would not change it if run again	50.00%	7
Total	**100%**	**14**

Table 4: Timing of any changes to CONTENT OF THE MATERIAL

Discussion

From the results, it is clear that evaluation was a frequent activity in the design and delivery of the sampled initiatives (36.7 reported incidents by any method at any time per initiative). The time-distribution of the evaluation incidents indicates incremental evaluation was happening for these projects.

Generally, the respondents in this study did not report acting on evaluations with the same high frequency that they carried these out. This may represent a missed opportunity to use these evaluations.

Thus, while some degree of incremental evaluation was observed, results suggest that this evaluation was not being applied iteratively. There may be opportunity for more flexible and tailored student-learning programmes if libraries examined the rationale and methodology of agile software development methods, and acted on evaluation more cyclically.

As can be seen from the pie chart below (Figure 3), even allowing for adjustment due to ease/possibility, LIBRARY STAFF (18% of all *possible/not very difficult* evaluations) was evaluated substantially less frequently than the other three aspects, which were within a range of 27-29%. Given that aspects of “library staff” are, presumably, the most easily examined, and theoretically most easily changed, it is noteworthy that this was the least evaluated aspect.

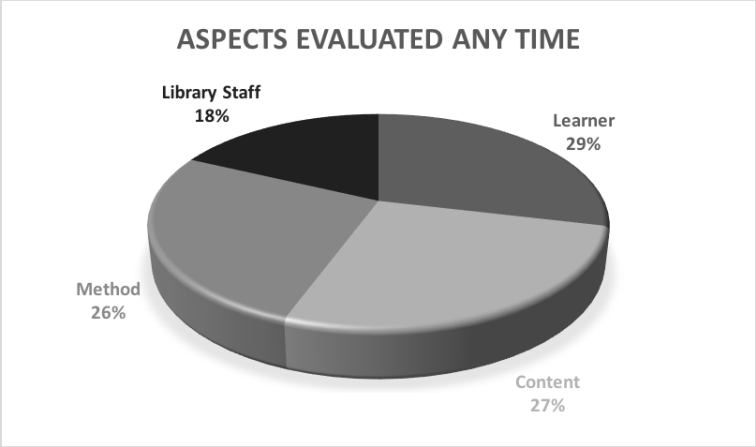

Figure 3: Aspects evaluated any time (weighted totals)

Timing of evaluation

This study sought to clarify more than just whether evaluation happened and how often, to include when it was primarily occurring in the design/delivery cycle. Iterative evaluative methods in software design involve frequent evaluation at all stages from design to delivery, and often even involve evaluation and amendment after the product is delivered to the client. The initiatives studied show this pattern, of regular evaluation at all points, in only a limited way.

The majority of evaluations (47%) happened after there was a chance to change what was happening for students, and a further 32% before the user experience could have been measured. There was less evaluation at the point at which both student impacts could be known, where changes could be made to improve these.

A fuller picture is revealed when considering evidence of when and how evaluation was applied, in relation to the stage at which it took place.

A secondary, less extensively-investigated part of this survey examined the points at which evaluation was used to make changes to the initiative. In software development that uses iterative and incremental design methods, evaluation is not only completed frequently at all stages of the project, but the information obtained is used to improve the design and methodology, and often results in a change to the goals of the project.

Use of evaluation to alter outcomes

Most evaluation had no impact on planned outcomes at all, which were set before development and left unchanged. One responded that the desired outcomes were not considered at any point. Of the remaining 93% of respondents, all of them had decided on the desired outcomes before the initiative was delivered: 69% “as part of the initial initiative brief”, and 31% “during development of the initiative”. This is despite 68% of all reported incidences of evaluation happening after delivery had commenced.

This replicates the rigid waterfall model of software development, which historically resulted in less flexible outcomes that were less tailored to user needs. This is what agile software development methodology sought to overcome with iterative evaluation and ongoing alteration of outcomes in response to this. Given the effort and time taken to evaluate, there may have been scope for these libraries to improve their practices by reconsidering and altering outcomes after the initial brief was decided.

Incremental but not iterative evaluation of method and content

As can be seen in Figure 2 above, roughly one third of all evaluations of method and content took place before delivery, one third happened more than 24 hours after delivery, while the remaining one third occurred either during or within 24 hours of the delivery. This was different to the evaluation pattern for the other two aspects, library staff and learner, where generally there was a lower proportion of evaluation more than 24 hours afterward, and a greater proportion of evaluation during the initiative.

The timing pattern for method and content suggest incremental evaluation. At three time points, there were equal incidences of evaluation. It is acknowledged that this does not indicate the depth and thoroughness of each incident, just that there was an incident of evaluation. This does suggest, however, that numerically at least, there was incremental evaluation.

As with outcomes, however, alteration to method and content in response to evaluation also represented very little change before or during delivery. For the student learning initiatives, 292 incidents of evaluation of content or method happened across all time periods. Half of these happened before the end of delivery, however only one alteration to content or method was reported during this period. Just one respondent reported changing content during or before the programme was run, while none reported changes to method during this time.

Despite the effort and time spent in evaluation, for over half of the projects, respondents indicated that no future changes to content or method were planned, should the project run again. In contrast to the 50% that would not alter the content if the project were to be run again, 57% of the respondents reported that they would not alter the method.

So, while evaluation was indeed happening before and during the project, the results of this were in all likelihood not being used within this time period. There is, of course, the possibility that evaluation verified that the content and method could not be improved upon, but given that library projects probably have the same level of fallibility and design flaws as other enterprises, this is unlikely. A more likely alternative explanation is that there was underreporting by respondents of subtle alterations made in response to less formal evaluation methods. For example, simplifying language in an explanation of a process during classroom delivery in response to discussion with a tutor during a break may not have been recorded as change of delivery method.

Even taking into account these other possible explanations for low reporting of changes to content and method of the initiative, the findings do suggest that *formalised* use of iterative evaluation to improve content and method in academic library student learning initiatives was not part of the developmental method for these projects.

The results indicate that while incremental evaluation is happening, there is a gap in the developmental methodology where library staff possibly could learn applicable techniques from iterative evaluation methods used in agile software development to improve their services.

Can incremental and iterative evaluation be useful for libraries?

As outlined above, Jim Highsmith, one of the original authors of the "Manifesto for Agile Software Development", identified the type of project that benefits best from agile "practices and principles" as "any project that faces uncertainty, complexity, volatility and risk" (Jackson, 2012, p. 61).

It is possible that some student learning initiatives in academic libraries are not complex or sufficiently risky for agile methodologies to be best used. Further research evaluating the scale and risk of library projects, and whether this makes them more or less likely to apply iterative evaluation, would be interesting. Certainly, unsuccessful student learning initiatives would involve a risk of wasting staff time and damaging the library's reputation or goodwill, so it is possible that agile methodology might offer some ways to reduce this risk and improve the projects. It is also clear that the involvement of students in the process brings complexity and uncertainty to these projects, again suggesting that there may be useful learning from applying agile methodology.

Further investigation into whether agile methodologies can be effectively applied to learning design in libraries could be informed by the present research, which shows that incremental evaluation was used, but that iterative evaluation appears not to have been. In considering this, two questions may be raised. How can libraries be encouraged to be more open to an experimental approach to developing new learning programs? How can an agile, iterative approach to programme development become a more common practice in libraries? Such questions, the answers to which are beyond the scope of this paper, open avenues for further research.

Conclusion

This research sought to provide a key piece of information in a wider discussion of whether incremental and iterative evaluation could be useful in libraries. It sought to discover whether there was evidence that incremental and iterative evaluation was already being used in libraries.

The results of this study show that, although some academic libraries took opportunity to evaluate the project considered at many points in the development/delivery cycle, there was far less evidence of use of this evaluation to alter goals, methodology or content after the initial conceptualisation of the design.

It is clear that among this sample of Australian academic libraries that delivered student learning projects there was evidence of frequent evaluation, which could be described as “incremental”. Equally apparent was that while there was a lot of evaluation happening, there was not as much response and change happening to the programmes at around the same time as the evaluations happened. This suggests there is scope for iterative evaluation to be considered as a way to improve delivery of learning initiatives to students by academic libraries.

References

Arimoto, M.M., Barroca, L., Barbosa, E.F., 2016. AM-OER: An Agile method for the development of open educational resources. Informatics in Education. 15 (2), 205–233. https://doi.org/10.15388/infedu.2016.11

Association of College and Research Libraries, 2016. Framework for Information Literacy for Higher Education. ACRL. http://www.ala.org/acrl/standards/ilframework

Baruch, Y., 1999. Response rate in academic studies--a comparative analysis. Human Relations. 52 (4), 421–438. https://doi.org/10.1023/A:1016905407491

Baruch, Y., Holtom, B.C., 2008. Survey response rate levels and trends in organizational research. Human Relations. 61 (8), 1139–1160. https://doi.org/10.1177/0018726708094863

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., 2001. Principles behind the Agile Manifesto. http://agilemanifesto.org/principles.html

Bober, C., Poulin, S., Vileno, L., 1995. Evaluating library instruction in academic libraries. The Reference Librarian. 24 (51-52), 53–71. https://doi.org/10.1300/J120v24n51_07

Booth, C., 2011. Reflective teaching, effective learning : instructional literacy for library educators. ALA Editions, Chicago.

Brooks, F.P., 1995. The mythical man-month. Addison-Wesley, Reading, Massachusetts.

Carlo Bertot, J., Snead, J.T., Jaeger, P.T., McClure, C.R., 2006. Functionality, usability, and accessibility: Iterative user-centered evaluation strategies for digital libraries. Performance Measurement and Metrics. 7 (1), 17–28. https://doi.org/10.1108/14678040610654828

Carlson, R., Turner, R., 2013. Review of Agile case studies for applicability to aircraft systems integration. Procedia Computer Science. 16, 469–474. https://doi.org/10.1016/j.procs.2013.01.049

Cervone, H.F., 2014. Improving strategic planning by adapting Agile Methods to the planning process. Journal of Library Administration. 54 (2), 155–168. https://doi.org/10.1080/01930826.2014.903371

Cervone, H.F., 2011. Understanding agile project management methods using Scrum. OCLC Systems & Services: International Digital Library Perspectives 27 (1), 18–22. https://doi.org/10.1108/10650751111106528

Chang, M., 2010. An Agile approach to library IT innovations. Library Hi Tech, 28 (4), 672–689. https://doi.org/10.1108/07378831011096303

Cockburn, A., 2004. Crystal Clear: A human-powered methodology for small teams. 1st ed. Addison-Wesley Professional, Boston.

Connaway, L., Radford, M., Powell, R., 2016. Research methods in Library and Information Science, 6th ed. Pearson Education, New York, N.Y.

Council of Australian University Librarians, 2014. CAUL Constitution. http://www.caul.edu.au/content/upload/files/caul-doc/constitution.pdf

Edin Grimheden, M., 2013. Can agile methods enhance mechatronics design education? Mechatronics. 23 (8), 967–973. https://doi.org//10.1016/j.mechatronics.2013.01.003

Farkas, M.G., Hinchliffe, L.J., Houk, A.H., 2015. Bridges and barriers: factors influencing a culture of assessment in academic libraries. College and Research Libraries, 76 (2), 150–169. https://doi.org/10.5860/crl.76.2.150

Gallant, J.W., Wright, L.B., 2014. Planning for iteration-focused user experience testing in an academic library. Internet Reference Services Quarterly, 19 (1), 49–64. https://doi.org/10.1080/10875301.2014.894954

Gustavson, A., Whitehurst, A., Hisle, D., 2011. Laying the information literacy foundation: a multiple-media solution. Library Hi Tech, 29 (4), 725–740. https://doi.org/10.1108/07378831111189796

Hufford, J.R., 2013. A Review of the Literature on Assessment in Academic and Research Libraries, 2005 to August 2011. portal: Libraries and the Academy, 13 (1), 5–35. https://doi.org/10.1353/pla.2013.0005

Jackson, M., 2012. Agile: a decade in. PM Network, 26 (4), 58–62. https://www.pmi.org/learning/library/agile-manifesto-software-development-impact-2284

Jacobs, M.A., Graham, A.L., 2016. Iterative development and evaluation methods of mHealth behavior change interventions. Current Opinion in Psychology 9, 33–37. https://doi.org/10.1016/j.copsyc.2015.09.001

Kaplowitz, K., Kaplowitz, J.R., 2012. Transforming information literacy instruction using learner-centred teaching. Facet Publishing, London.

Larman, C., Basili, V.R., 2003. Iterative and incremental developments: a brief history. Computer 36 (6), 47–56. https://doi.org/10.1109/MC.2003.1204375

Lei Hsieh, M., Holden, H.A., 2010. The effectiveness of a university's single-session information literacy instruction. Reference Services Review, 38 (3), 458–473. https://doi.org/10.1108/00907321011070937

Levy, S., 1981. Hackers: heroes of the computer revolution. Penguin.

Manzari, L., Trinidad-Christensen, J., 2006. User-Centered design of a web site for library and information science students: Heuristic evaluation and usability testing. Information Technology and Libraries, 25 (3), 163–169. https://doi.org/10.6017/ital.v25i3.3348

Oakleaf, M., 2009. The information literacy instruction assessment cycle: a guide for increasing student learning and improving librarian instructional skills. Journal of Documentation, 65 (4), 539–560. https://doi.org/10.1108/00220410910970249

Oakleaf, M., Hinchliffe, L.J., 2008. Assessment cycle or circular file: do academic librarians use Information Literacy assessment data? Presented at the 2008 Library Assessment Conference. Building effective, sustainable, practical assessment., Seattle, Washington, pp. 159–164. http://meganoakleaf.info/circularfile.pdf

Oakleaf, M., Kaske, N., 2009. Guiding questions for assessing information literacy in higher education. portal: Libraries and the Academy, 9 (2), 273–286. http://dx.doi.org/10.1353/pla.0.0046

Ondrusek, A., Dent, V.F., Bonadie-Joseph, I., Williams, C., 2005. A longitudinal study of the development and evaluation of an information literacy test. Reference Services Review, 33 (4), 388–417. https://doi.org/10.1108/00907320510631544

Osorio, J.A., Chaudron, M.R.V., Heijstek, W., 2011. Moving from waterfall to iterative development: an empirical evaluation of advantages, disadvantages and risks of RUP. Software Engineering and Advanced Applications (SEAA), 2011 37th EUROMICRO Conference, 453–460. https://doi.org/10.1109/SEAA.2011.69

Poppendieck, M.B., Poppendieck, T.D., 2013. Lean software development: an agile toolkit. Addison Wesley, Boston.

Salisbury, F., Ellis, J., 2003. Online and face-to-face: Evaluating methods for teaching information literacy skills to undergraduate arts students. Library Review, 52 (5), 209-217. https://doi.org/10.1108/00242530310476715

Savage, D., Piotrowski, P., Massengale, L., 2017. Academic librarians engage with assessment methods and tools. portal: Libraries and the Academy, 17 (2), 403–417. https://doi.org/10.1353/pla.2017.0025

Schwaber, K., Beedle, M., 2001. Agile software development with Scrum, 1st ed. Pearson, Upper Saddle River, NJ.

Somerville, M.M., Brar, N., 2009. A user-centered and evidence-based approach for digital library projects. The Electronic Library, 27 (3), 409–425. https://doi.org/10.1108/02640470910966862

Swoger, B., 2011. Closing the assessment loop using pre- and post-assessment. Reference Services Review, 39 (2), 244–59. https://doi.org/10.1108/00907321111135475

Wong, C., 1984. A successful software development. IEEE Transactions on Software Engineering, SE-10 (6) 714–727. https://doi.org/10.1109/TSE.1984.5010300

Yocum, K.A., 2015. Designing creativity: using agile principles in instructional design for online learning. Dissertation, Capella University.

Appendix One - Calculation of weighting of reported incidents of evaluation by time point, by aspect

This research sought to identify where there was a design decision to evaluate at a given time point. For evaluation methods that were possible or relatively easy, it is reasonable to conclude that the decision to use this method at a particular time was a design decision (which would include decisions not to evaluate due to perceived lack of time or resources).

Some raw scores, however, included time points where the reason that a particular method was not used was more likely to be because that method was not possible or was very difficult. To allow better comparison between the aspects, some degree of control for ease and possibility needed to be applied to the results.

Applying an adjustment rate to the raw score was preferable to keeping and comparing only those evaluation methods that were *possible/not very difficult* for every single listed data point. Excluding evaluation methods only applicable to three aspects or two time points may well have excluded a very commonly used/appropriate method for one aspect/time point, purely because this method did not suit another aspect/time point. This approach would cause a worse skew to the raw scores than adjusting them, as chosen, according to possibility/ease.

Each of the four "aspect" matrices (LEARNER, CONTENT, METHOD, LIBRARY STAFF) were analysed by the researchers to determine whether each of the 14 evaluation methods at each of the four time points (*before, during, within 24 hours, more than 24 hours*) were "impossible", "very difficult", "difficult", "neutral", "easy" or "very easy". Those that were "impossible" or "very difficult" were excluded using a red cross.

The tables below show the evaluation points that were excluded.

	Method	LEARNER	before	during	within 24 hours	more than 24 hours after	Number of opportunities
1	Survey of potential participants		✓	✗	✗	✗	1
2	Survey of actual participants		✓	✓	✓	✓	4
3	Conversation with actual participant(s)		✓	✓	✓	✓	4
4	Conversation with academic staff		✓	✓	✓	✓	4
5	Reflection by library staff member responsible for initiative delivery		✓	✓	✓	✓	4
6	Formal observation and feedback by other library staff		✓	✗	✓	✓	3
7	Informal observation and feedback by other library staff		✓	✓	✓	✓	4
8	Small-scale pilot version of planned activity		✓	✗	✗	✗	1
9	Research into other similar projects (eg. web searching, literature review, informal discussion)		✓	✗	✓	✓	3
10	Discussion with an external community of practice		✗	✗	✓	✓	2
11	Computer generated analytics (eg. automatic count of site hits, number of logins, percentage completion rate)		✓	✓	✓	✓	4
12	Activity evaluating participant skill		✗	✓	✓	✓	3
13	Social media backchannel		✓	✓	✓	✓	4
14	Other (specify)		✓	✓	✓	✓	4
	NUMBER OF OPPORTUNITIES		12	9	12	12	

	Method	CONTENT	before	during	within 24 hours	more than 24 hours after	Number of opportunities
1	Survey of potential participants		✓	✗	✗	✗	1
2	Survey of actual participants		✓	✓	✓	✓	4
3	Conversation with actual participant(s)		✓	✓	✓	✓	4
4	Conversation with academic staff		✓	✓	✓	✓	4
5	Reflection by library staff member responsible for initiative delivery		✓	✓	✓	✓	4
6	Formal observation and feedback by other library staff		✓	✓	✓	✓	4
7	Informal observation and feedback by other library staff		✓	✓	✓	✓	4
8	Small-scale pilot version of planned activity		✓	✗	✗	✗	1
9	Research into other similar projects (eg. web searching, literature review, informal discussion)		✓	✗	✓	✓	3
10	Discussion with an external community of practice		✓	✗	✓	✓	3
11	Computer generated analytics (eg. automatic count of site hits, number of logins, percentage completion rate)		✗	✓	✓	✓	3
12	Activity evaluating participant skill		✓	✓	✓	✓	4
13	Social media backchannel		✓	✓	✓	✓	1
14	Other (specify)		✓	✓	✓	✓	2
	NUMBER OF OPPORTUNITIES		13	10	12	12	147

	Method	METHOD	before	during	within 24 hours	more than 24 hours after	Number of opportunities
1	Survey of potential participants		✓	✗	✗	✗	1
2	Survey of actual participants		✓	✓	✓	✓	4
3	Conversation with actual participant(s)		✓	✓	✓	✓	4
4	Conversation with academic staff		✓	✓	✓	✓	4
5	Reflection by library staff member responsible for initiative delivery		✓	✓	✓	✓	4
6	Formal observation and feedback by other library staff		✓	✓	✓	✓	4
7	Informal observation and feedback by other library staff		✓	✓	✓	✓	4
8	Small-scale pilot version of planned activity		✓	✗	✗	✗	1
9	Research into other similar projects (eg. web searching, literature review, informal discussion)		✓	✗	✓	✓	3
10	Discussion with an external community of practice		✓	✗	✓	✓	3
11	Computer generated analytics (eg. automatic count of site hits, number of logins, percentage completion rate)		✗	✓	✓	✓	3
12	Activity evaluating participant skill		✓	✓	✓	✓	4
13	Social media backchannel		✓	✓	✓	✓	4
14	Other (specify)		✓	✓	✓	✓	4
	NUMBER OF OPPORTUNITIES		13	10	12	12	

	Method	LIBRARY STAFF	before	during	within 24 hours	more than 24 hours after	Number of opportunities
1	Survey of potential participants		✓	✗	✗	✗	1
2	Survey of actual participants		✓	✓	✓	✓	4
3	Conversation with actual participant(s)		✓	✓	✓	✓	4
4	Conversation with academic staff		✓	✓	✓	✓	4
5	Reflection by library staff member responsible for initiative delivery		✓	✓	✓	✓	4
6	Formal observation and feedback by other library staff		✓	✓	✓	✓	4
7	Informal observation and feedback by other library staff		✓	✓	✓	✓	4
8	Small-scale pilot version of planned activity		✓	✗	✗	✗	1
9	Research into other similar projects (eg. web searching, literature review, informal discussion)		✓	✗	✓	✓	3
10	Discussion with an external community of practice		✓	✗	✓	✓	3
11	Computer generated analytics (eg. automatic count of site hits, number of logins, percentage completion rate)		✗	✗	✗	✗	0
12	Activity evaluating participant skill		✗	✗	✗	✗	0
13	Social media backchannel		✓	✓	✓	✓	4
14	Other (specify)		✓	✓	✓	✓	4
	NUMBER OF OPPORTUNITIES		12	7	9	9	

Appendix One Figure 1: Data collection points excluded as impossible or very difficult

This process identified the data collection points that could be fairly compared between the aspects, that is, the total number of *possible/not very difficult* evaluations by any method at each time point/aspect (shown in the "NUMBER OF OPPORTUNITIES" figure above).

This figure was then translated into a weighting ratio, which was calculated by dividing the number of surveyed collection points (always 14) by the number of *possible/not very difficult* collection points (which varied according to the number of red crosses in the column). So, in the "Library Staff" table above, for example, the "before" collection point had 12 of 14 *possible/not very difficult* collection points, resulting in an adjustment rate of 14/12.

This meant that raw totals where 14/14 collection points were *possible/not very difficult* would not be adjusted at all, while the lower the number of *possible/not very difficult* collection points there were, the more the raw total would be increased to compensate.

The table below shows the adjustment rates for each time point/aspect, which is the NUMBER OF OPPORTUNITIES divided by 14.

		TIMING OF EVALUATION			
		before	during	within 24 hours	more than 24 hours after
ASPECT EVALUATED	Learner	1.166666667	1.555555556	1.166666667	1.166666667
	Content	1.076923077	1.4	1.166666667	1.166666667
	Method	1.076923077	1.4	1.166666667	1.166666667
	Library Staff	1.166666667	2	1.555555556	1.555555556

Appendix One Table 1: Weighting of reported incidents: number of collection points divided by number of opportunities for each time point/aspect

The raw totals of reported data collections at each time point/aspect were then adjusted by the figure in the “Weighting” table above. This figure has then been rounded to the nearest two decimal places. This table is discussed in the “RESULTS” section of this paper.