In *Lecture Notes in Computer Science - Advances in Web-Based Learning*,
Volume 3143/2004, Springer-Verlag Heidelberg, pp. 11-18

The Agile Teaching/Learning Methodology and its e-Learning Platform

Andy Hon Wai CHUN

Department of Computer Science
City University of Hong Kong
Tat Chee Avenue, Kowloon
Hong Kong SAR
andy.chun@cityu.edu.hk

Abstract. The Agile Teaching/Learning Methodology (ATLM) is a teaching/learning methodology designed for higher-education based on the best practices and ideas from the field of software engineering and leveraging upon concepts from agile software methodologies. Although ATLM was designed using concepts borrowed from software engineering, the methodology itself can easily be applied to a wide variety of courses that might require agility in teaching and learning. This paper explains the objectives behind ATLM and the process architecture of the methodology. ATLM emphasizes agility, communication and the learning process. The paper also presents the e-learning platform we have developed to support this ATLM approach to teaching/learning and the technologies behind this platform. The ATLM e-learning platform makes use of a number of modern collaboration and knowledge sharing technologies such as blogging, commenting, instant messaging, wiki and XML RSS.

Keywords: teaching methodology, learning methodology, e-learning platform.

1 Introduction

Agile Teaching/Learning Methodology (ATLM) is systematic approach to teaching/learning that has been successfully applied to the teaching of several Computer Science courses at the City University of Hong Kong [1] for a number of years. Although used for teaching technology-related courses, we believe the methodology itself is general enough to be applied to other disciplines as well. Many other disciplines share the same teaching/learning objectives and goals that are promoted by ATLM. For example, teaching must be agile to cope with changing and diverse learning needs. Learning must be agile to cope with changing research, business, and technology environments. ATLM also encourages communication, knowledge sharing, and the learning process to nurture self-learning individuals. As the ATLM name implies, it is a methodology for teaching as well as a methodology for learning. Teachers need a well-defined approach to teaching just as much as a student needs a well-defined approach to learning. Teaching and learning, of course, go hand-in-hand. ATLM is a balanced methodology that supports both sides of the equation.

Software Engineering (SE) practitioners have long been well aware of the importance of adopting a well-defined methodology for software development [2]. Software development is a highly complex process and it makes sense to have well-defined steps, tasks and plans as well as an understanding of the dynamics behind the whole development lifecycle and how to deal with expected and unexpected changes, problems and risks.

It turns out that the teaching process is, in many ways, very similar to the software development process. It involves multiple parties with different objectives (sometimes conflicting), a very tight schedule to get things done, a fixed deadline, limited resources and a lot of expected/unexpected changes along the way. Both the teaching and software development processes require detailed planning/scheduling, tracking and management with continuous assessment and feedback from all parties. Getting a software project done correctly and on time is not easy. Making a sure a course is taught properly and on schedule can also be challenging sometimes.

In the past decade, there has been a gradual trend towards favoring a set of "light weighted" software development methodologies called agile processes [3]. These agile processes all follow the same principles as defined in the Agile Manifesto [4] that basically states that for a project to be successful we should value individuals and their interactions rather than rigidly following a process, working software rather than documents and specifications, communication rather than contract negotiation, and responding to changes rather than just following a plan. Agile methodologies put people first and are self-adaptive.

We see a lot of parallels between agile software development and modern teaching and learning. Agile Teaching/Learning Methodology values students/teachers and their interactions rather than a particular approach to teaching/learning, working knowledge rather than rote-learning, communication rather than negotiation, and responding to changes rather than just following a schedule. ATLM emphasizes teacher student communication and stresses the importance of being self-adaptive to cater to changing needs.

2 Key Characteristics

Inspired by these modern agile methodologies for software development, we designed ATLM with three key characteristics – agility, extreme, and independence.

- **Agility** - By agility, we mean the ability for the teacher to quickly adapt and change course pace and possibly structure to suit the needs and abilities of the students. The main objective for teaching is to help students learn. Each and every student is unique. Their learning needs are also unique. Not all students in a class will be able to learn at the same rate or in the same manner. Each time a course is offered, it should not be taught exactly the same way as the students in each class will be different with different learning needs. A teacher will need to ensure that different learning needs are catered to rather than just following a predefined plan.

Another way to describe this is "in-sync teaching" – making sure the whole class is in-sync with the material being taught. This is particularly important when there is a mixture of students with different academic backgrounds in the class. We do not want to confuse learning rate with learning capacity. As educators, we need to be sensitive to and facilitate the needs of students from a diverse academic background. It is important to be able to patiently shepherd and encourage weaker students, while showing equal respect and openness.

Agility also means making sure the class, both the teaching and learning aspects, are constantly on track from day one. A good teacher, of course, will need to know the subject well. But, he/she must also know the students well and how they are progressing. Agility enables a self-adaptive teaching/learning process, which adapts to the specific learning needs of each class.

ATLM achieve this type of agility by maintaining a high degree of communication, interaction and feedback between the student and teacher. Teachers need to know how the students are doing and how they like/dislike the teaching approach. Students need to know what is expected from them. ATLM uses a variety of technologies to open up many channels of communication to achieve agility. Details of our e-learning platform will be explained later in the paper.

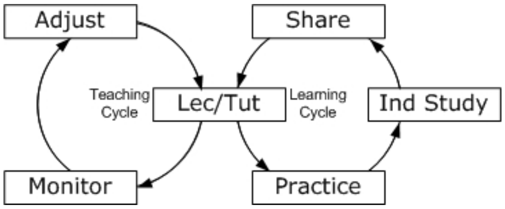
- **Extreme** - By extreme, we mean if something is good for teaching/learning, we go to the extreme and do it constantly [5]. If giving feedback to students on their learning is good, we should give feedback each class. Conversely, if getting feedback from students is good, we should allow students to give feedback whenever they want. If students learn better by teaching, they should be allowed to teach and share their knowledge constantly.

  ATLM encourages teachers to solicit student feedback on their teaching constantly and from day one. This should not be a formal teaching assessment but rather an informal and possibly anonymous feedback from the student with the main intent to guide and fine-tune the teaching approach. Formal teaching assessments are usually done towards the end of the semester. By the time teachers get results from them, it will be too late to do anything to help improve the teaching for that class. The ATLM e-learning platform provides many different options and means to solicit feedback.

- **Independent** - One of the aspects of ATLM is to train students on "learning the learning process" and not just learning the course content. One of the objectives of ATLM is to gradually take the role of the teacher "out of the loop." With each iteration in the ATLM teaching/learning cycle, the student grows more confident of self-learning so that ideally at the end of the course, the students will be confident in continuing the learning process on their own. This is extremely important for disciplines that are constantly changing, such as Computer Science. Some of the technologies student learn in school can become obsolete in a short time; students need to constantly update themselves and learn on their own. Empowering students with skills for life-long learning is important aspect of ATLM.

3 Iterative Teaching/Learning Cycles

Like other modern software development methodologies [2], ATLM is also an iterative methodology. This means the cycles of the methodology are performed over and over again in iterations; each iteration may progressively have slight variations to be self-adaptive. For teaching, each lecture or lecture/tutorial combination is one iteration. In ATLM, since we are dealing with both teaching and learning, there are two cycles that operate in parallel in each iteration – one for the teacher and one for the student:



The Teaching Cycle (on the left of the diagram) is for the teacher to follow, while the Learning Cycle (on the right) is for students. Both the Teaching and Learning Cycles share the Lecture/Tutorial task:

- **Lecture/Tutorial** – this is the standard lecture and tutorial components of a class. In ATLM, we assume each iteration is the time between two lectures; usually a week. The role of the teacher-led lecture/tutorial will be greater at the beginning of the course and will gradually diminish towards the end when the students are confident in the course material and can begin to learn independently on their own, using skills and techniques learned in the course.

In addition, the Teaching Cycle has the following tasks:

- **Monitor** – As one of the key characteristics of ATLM is agility in meeting student needs, the teacher must constantly monitor student progress as well as feedback from students on their own teaching progress/performance. Student progress can easily be monitored, for example, by providing a simple weekly quiz that might or might not be counted as part of the official student assessment. Quiz results should be provided to students immediately after the class/tutorial. Statistics on quiz results will help students understand learning expectations and also identify weaknesses that need more work. Some teachers do not like giving weekly quizzes, even if they are informal quizzes. The result is a less agile class. Teachers will have to wait until the first assignment has been collected and graded to see how well students are doing. This may take several weeks. For a semester, this latency is too slow and not efficient.

  In addition to student performance, the teacher will also need to monitor his/her own teaching performance as perceived by the current group of students. This can be done through feedback forms, surveys as well as forums that students are encouraged to participate on an on-going basis. The feedback can be anonymous. The main point is for the students to understand that these are not teacher assessment exercises but rather a way to allow the teacher to adapt his/her teaching approach to cater to current students learning needs. As you can see, feedback is taken to the "extreme" in ATLM and is done constantly at each iteration.

- **Adjust/Adapt** – Once student and teaching performance are both understood, the teacher should immediately make any necessary adjustments to the course plan, schedule or content as well as coursework to help ensure students are "in-sync" with their learning. Changes are necessary for agility.

All these three steps are done during each Teaching Cycle. For the Learning Cycle, we have in addition:

- **Practice** – this is the standard coursework component of a class. The nature of the coursework will of course depend on the course being taught. However, ATLM encourages giving students assignments that help reinforce a working knowledge of the course material instead of simply rote-learning. Assignments should also be open-ended with plenty of room for creativity. The creativity aspect also adds an element of competition among the students to create more interesting solution to the assignment problem.
- **Independent Study** – One of the aspects of ATLM is to promote learning skills and prepare students for lifelong learning. Tasks should be given to gradually guide students in learning the learning process and understanding where to find resources to support their learning. ATLM encourages knowledge sharing and collaboration among students as a catalyst for self-learning. Getting students ready for lifelong learning is an important component of ATLM.
- **Knowledge Sharing** – Sharing knowledge is an important part of learning! At each teaching/learning iteration, some time should be allocated for students to talk and share what they have learned during their own studying/research. For example, students may be asked to simply find something interesting from the Web that is related to what is being taught in class and to share that with other students. It should not be a formal presentation and there is no PowerPoint to prepare for. Instead, the student might just open a Website and explains what he thought was interesting, what new concepts he learned and how it relates to the course.

  Letting students "teach" greatly enhances their learning experience. Firstly, they learn from others. Secondly, they learn when they prepare for their own presentations. But, most importantly, through the whole process they get to learn one of the most important skills of their life – how to learn on their own!

  The classic Learning Pyramid confirms this approach [6]. It charts the average retention rate for various methods of teaching. These retention percentages represent the results of research conducted by National Training Laboratories in Bethel, Maine in 1994. According to that chart, lecture, the top of the pyramid, achieves an average retention rate of only 5%. On the opposite end of the scale, the "teach others/immediate use" method achieves an average retention rate of 90%! ATLM believes that knowledge sharing or "teaching others" is a critical tool in helping students understand the learning process and helps prepare them for lifelong learning.

In addition, to sharing in class, ATLM encourages students to share online through the use of several modern collaboration and knowledge management tools such as blogs [7, 8] and wikis [9]. Blogging is currently probably the most common way of sharing knowledge on the Internet. A blog is like an online diary with entries on different topics. The power behind blogging is that others can join in the discussion through commenting [10] or trackback [11] techniques as well as subscribing to blogs. Wiki, on the other hand, is a simple content management system that allows anyone with appropriate logon privileges to add/change and contribute to Web content on-the-fly. Both blog and wiki are powerful technologies that greatly streamline and simplify the process of collaboration and knowledge sharing.

4 Teaching/Learning Best Practices

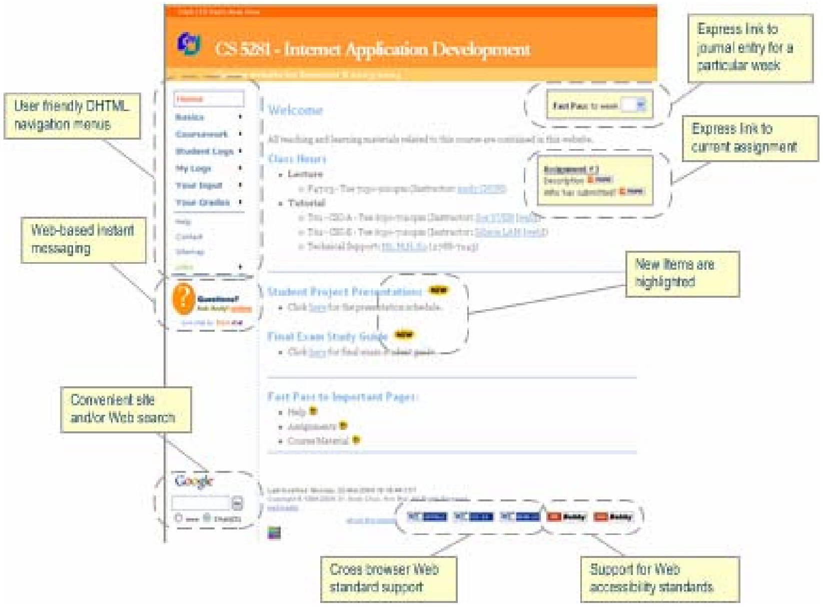
There are numerous teaching/learning best practices that can be practiced with ATLM. However, ATLM particularly promotes and emphasizes the following as part of the methodology:

- **Learn by Sharing** – ATLM makes use of the fact that students learn over an order of magnitude better if they also participate in the teaching process. ATLM facilitates this through knowledge sharing exercises.
- **Teach How to Learn** – ATLM emphasizes that, in addition to course content, it is important to also teach the learning process. ATLM does this through guided and targeted independent study tasks with knowledge sharing and collaboration as motivation.
- **Feedback is Good** – Feedback is what makes ATLM agile. Without feedback, the teacher will not be able to improve the course delivery and teaching. Without feedback, students will not know if their work is on track and inline with expectations. ATLM facilitates feedback through informal weekly quizzes and feedback forms, surveys and online comments.

5 The ATLM e-Learning Platform

Our e-learning platform makes use of an integrated set of innovative technologies to support the highly dynamic and agile mode of teaching as required by ATLM. The platform leverages on some of the newest software technologies for knowledge sharing and collaboration. This platform can supplement more traditional technologies such as forums and blackboards. The following diagram shows the front page of one of the course websites that uses the ATLM e-learning platform:

Firstly, the e-learning platform makes use of open technologies, standards and Web-based services. Most of these technologies and services are free. For example, the whole platform complies to XHTML 1.1 and CSS 2.0 Web standards to ensure the platform can be displayed on all the popular browsers. In addition, it meets W3C as well as Bobby 508 and AAA accessibility recommendations. The following is a list of some of the technologies and services used by the ATLM e-learning platform:


CS 5281 - Internet Application Development
Home
Basics
Coursework
Student Logs
My Logs
Your Input
Your Grades
Welcome
Class Hours
Lecture
Tutorial
Questions?
Student Project Presentations
NEW
Final Exam Study Guide
NEW
Fast Pass to Important Pages:
Help
Assignments
Course Material
Google
Bobby
Bobby
Express link to journal entry for a particular week
User friendly DHTML navigation menus
Express link to current assignment
Web-based instant messaging
New items are highlighted
Convenient site and/or Web search
Cross browser Web standard support
Support for Web accessibility standards

- **Blogging** – Blogging is the newest trend on the Internet. It is like an online diary but a lot more. It is tightly linked to other web pages. It allows others to comment entries as well as subscribe to diaries. And there are specialized search engines to support blog. Blogging is simple and can even be done with a mobile phone! In ATLM students use blogs to share what they have learned by writing a small entry for each interesting thing they have learned or found through research about the topic the class is covering.
- **Wiki** - Wiki is a popular Web-based collaborative tool. It allows content to be easily changed on-the-fly. ATLM uses Wiki to allow students to share information related to the course. A wiki can also be used for FAQ or to maintain resource links.
- **Commenting/Rating** – The platform makes liberal use of commenting technology. Comments can be attached to practically any entry on the course Web site. This greatly improves communication and reduces misunderstanding. In addition, ATLM encourages peer-review through commenting with ranking scores, similar to book ranking in amazon.com.
- **Instant Messaging** – The platform allows students to chat with the teacher online without needing to install any additional software. The convenience of Web-based chat further encourages and promotes communication.

The platform provides shared services that are available on all pages, services such as commenting, instant messaging, search, multi-lingual translations, and updated notifications via XML RSS.

6 Conclusion

We have presented a novel teaching/learning methodology – the ATLM, which we have successful used for teaching technical courses. The methodology focuses on the learning process to ensure students can continue to learn and update their knowledge even after the course has been completed. This is crucial for technical courses, since technologies change so rapidly and what students learn in class may quickly become outdated. However, at the same time, we feel the motivations and objectives behind ATLM might also make it useful for non-technical course content as well. So far, we have applied ATLM and the e-learning platform for several Computer Science courses on programming and software engineering with very good response from students, especially for part-time students who prefer Web-based interactions. We plan to continue to improve the e-learning platform and possibly convert it to a generic framework that can be used by other universities as well.

7 Acknowledgement

The work described in this paper was substantially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. 9040517, CityU 1109/00E). The work described in this paper was also partially supported by a grant from the City University of Hong Kong (Project No. 7001286 and 6980002).

References

1. The City University of Hong Kong, http://www.cityu.edu.hk
2. Jacobson, I., Booch, G., and Rumbaugh, J., *The Unified Software Development Process,* Addison-Wesley (1999).
3. Fowler, Martin, "The New Methdology," April 2003, http://www.martinfowler.com/articles/newMethodology.html
4. Manifesto for Agile Software Development, http://agilemanifesto.org/
5. Extreme Programming: A gentle introduction, http://www.extremeprogramming.org/
6. National Training Laboratories, "The Learning Pyramid," Workshop Materials, Bethel, Maine.
7. Blogger, http://www.blogger.com/about.pyra
8. Radio Userland, http://radio.userland.com/
9. What is Wiki, http://wiki.org/wiki.cgi?WhatIsWiki
10. Blogkomm, http://www.blogkomm.com/
11. Trackback Development, http://www.movabletype.org/trackback/