An Agile Approach to Systems Analysis and Design Teaching and Learning

Completed Research Paper

Liza Pieters
Midrand Graduate Institute
lizak@mgi.ac.za

ABSTRACT

Systems analysis and design (SAD) is a core part of systems development and the Information Systems curriculum (Gorgone et al, 2002; Bajaj et al, 2004). SAD is a complex process, with many techniques used to understand problems and design solutions. This paper investigates how an agile software development approach may be used for the teaching and learning of SAD. The research reviews and compares literature on agile teaching and learning (ATL) and the challenges of SAD teaching and learning (T&L). The ATL approach is used to address the challenges of SAD T&L. Accordingly, the paper proposes an agile SAD T&L approach. The main limitation of the study is that the proposed approach has not been tested. The most promising concept of the proposed agile SAD T&L approach is its flexibility and ability to adapt and adjust according to individual preferences and needs.

Keywords

Agile Teaching and Learning (ATL), Systems Analysis and Design (SAD), Agile methodology, Teaching and Learning (T&L).

INTRODUCTION

Systems development is a complex process which requires developers to understand the whole development life cycle and to adapt to planned and unexpected changes, problems and risks (Chun, 2004). Many different systems analysis and design methodologies exist, approaching the systems development process differently (Wood-Harper and Fitzgerald, 1982). In reality there is still a high rate of systems development failure which is often related to the inability of systems development approaches to adapt and cope with changes (Ferreira and Cohen, 2008). In response, the agile systems development approach emerged. The agile approach is tailored to cope with and respond to change and accepts that requirements of systems can change, and subsequently also the development of that system (agilemethodology.org, 2008).

Systems analysis and design (SAD) is core to systems development and an established part of Information Technology (IT) courses (Cybulski and Linden, 2000). “In an agile paradigm, every aspect of development is continually revisited throughout the lifecycle” (agilemethodology.org, 2008). This agile approach to systems development directly effects how the systems analysis and design phases are executed, but could it not also inform its teaching and learning? Teaching is after all quite similar to such a systems development approach (Greenbaum and Mathiassen, 1990: 526). “We never really know what the end result is going to be like and how it is going to be used” (Greenbaum and Mathiassen, 1990: 526).

This paper explores SAD teaching and learning (T&L) as well as agile teaching and learning (ATL). The aim is to determine if an ATL approach can address SAD T&L challenges.

SYSTEMS ANALYSIS AND DESIGN TEACHING AND LEARNING

In 1978, Spence and Grout suggested that systems analysis and design (SAD) become part of the Computer Science curriculum. In the IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems published by the Association for Information Systems, SAD forms a core part of the Information Systems (IS) curriculum.

“The struggle to teach the concepts of SAD to students who find it difficult to internalize either the concepts or the processes has been the subject of extensive scholarship spanning the information era” (Suscheck and Huff, 2007). Teaching students to understand the problem (analysis) and subsequently to come up with the best possible solution (design) (Albayrak, 2000) is challenging, especially if they have no systems development background (Suscheck and Huff, 2007).

Teaching SAD in the traditional lecturing way, place students in a passive receptive role, which results in students absorbing only about 30% of what the lecturer is explaining (Cybulski and Linden, 2000). A realistic approach to teaching SAD has
been proposed (Archer, 1985), and innovative approaches have been researched (Olfman and Bolstron, 1992). More common approaches to teaching SAD include using case studies, real life examples, simulations, tutors, close to real life projects, and teamwork (Cybulski and Linden, 2000). The common theme to these teaching approaches is to provide the students with opportunity to learn through application and experience, instead of the teacher being the main source of information. The aim in mind is to help students to make “the transition from textbook understanding of information systems to actual real world experience” (Wallace, 2005). Hands-on experience is in effect how practising system analysts improve their skills and application knowledge (Suscheck and Huff, 2007). The challenge is that good SAD requires in-depth project or case study understanding (Suscheck and Huff, 2007), fine technical skills and even a sound knowledge of business and management (Cybulski and Linden, 2000).

One of the biggest challenges however with using case studies and simulations or exercises based on real life systems, is that there is not one best way to arrive at the ultimate solution (Cybulski and Linden, 2000). In SAD there are different techniques, such as structured and object oriented modelling, that can be used to arrive at various solutions. According to (Gorgone et al, 2002) both structured and object oriented methodologies should be included in SAD courses as part of an IS curriculum. SAD is not as “precise as programming and not exact like Mathematics” (Smit, 2012), but rather requires creative thinking skills (Olfman and Bolstron, 1992). Therefore SAD is more difficult to teach (Cybulski and Linden, 2000) and equally difficult to learn (Smit, 2012). The students need to develop their own cognitive abilities to select a technique and judge when a solution could be considered good and when it could be improved. Olfman and Bolstron (1992) emphasised facilitation skills such as outcome thinking, group process and communications to improve SAD T&L.

As part of the SAD course, traditional and agile systems development methods are introduced, explained and contrasted (Bajaj et al, 2004). In 1982, Wood-Harper and Fitzgerald stated that “there exists a confusing array of approaches”. They also noted that these “approaches are not just simple alternatives, but that they seek to do different things” (Wood-Harper and Fitzgerald, 1982). While the traditional approaches emphasise a sequential process, the agile systems development approach attempts to provide a solution to system developers towards developing better systems by accepting and reacting to changes, adopting an “inspect-and-adapt” approach (agilemethodology.org, 2008). These are attractive qualities that could be used for SAD T&L.

AGILE TEACHING AND LEARNING

ATL is a relatively new approach being explored for its application potential in higher education. The agile systems development methodology is a “creative process that anticipates the need for flexibility and applies a level of pragmatism into the delivery of the finished product” (Rouse, 2007). The agility of the approach is attractive and holds many advantages for the education setting (Davey and Parker, 2010).

The need for an agile approach to T&L comes from a general need for education to respond to what is going on in the classroom as well as around and beyond it (Berry, 2012). Generally educators root all their planning in solid curriculum, but “the outcome is only successful in the way students apply the concepts and approaches within the realm of their experiences” (Greenbaum and Mathiassen, 1990). This requires education to accept that students are not just passive recipients of our teaching; they should rather be looked upon as contributing partners in the T&L experience (Berry, 2012). The T&L should therefore be learner-centred and flexible, where the individuals are taught rather than the planned out lessons (Berry, 2012).

ATL adopts the principles defined in the Agile Manifesto: The principles state that the focus should be on individuals and interactions, rather than processes and tools; on working software over documentation; on customer collaboration, not contract negotiation, and on response to change over following a plan (Agilemanifesto.org).

Based on these principles, Andy Chun (2004) developed the Agile Teaching/Learning Methodology (ATLM) approach that “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 is an iterative methodology that allows self-adaptive variations to be applied to iterations (Chun, 2004). These iterations involve the Teaching cycle and the Learning cycle. The Teaching cycle includes lecturing the standard lecture, monitoring the student progress and the feedback received from students, and then adapting by making the needed adjustments to the lesson in terms of schedule, content or coursework (Chun, 2004). While the Teaching cycle reiterates, the Learning cycle runs in parallel (Chun, 2004). The Learning cycle includes students practicing coursework through assignments or activities, promoting independent study to encourage lifelong learning, and then sharing their newly acquired knowledge (Chun, 2004).
This approach agrees with the claim that ATL is about starting at the beginning and not the end by “making use of what learners know already and building on that” (Berry, 2012). Berry (2012) suggests that the teacher should have immediate objectives that are immediately useful, instead of having a pre-planned sequence to a pre-determined destination.

**RESEARCH APPROACH**

This research is based on the analysis of qualitative data collected from existing literature. Qualitative data analysis is executed to determine if and how ATL can inform SAD T&L. The researcher believes that subjective realities influence systems development and therefore SAD T&L. There is no one right way to develop systems, but each system should rather be developed in its own changing context. The agile approach addresses these challenges by accepting change and adapting and adjusting development as needed. To apply a similar view to SAD T&L, the ATL approach is explored in terms of how it can address the challenges of SAD T&L.

The subjective interpretations, views and realities found in existing literature, combined with the experience of the researcher as an IS lecturer, are then used to propose a fresh approach to SAD T&L. The research aims to contribute to SAD T&L by providing qualitative data that could be used as a starting point for future qualitative and quantitative studies. On account of this research, the approach suggested is being tested by the researcher through practical application for further research.

**CONNECTING ATL WITH SAD TEACHING AND LEARNING**

In order to understand how to connect the dots between SAD and ATL, the similarities or connecting attributes are identified. It is important to revisit the fact that both traditional and agile methodologies form part of the SAD curriculum (Bajaj et al, 2004). While the agile methodology forms part of the SAD curriculum, it was also used to develop an ATL approach (Chun, 2004). Starting with Figure 1, the research attempts to determine if and how ATL can be applied to SAD T&L.

![Figure 1 merging the two concepts](image-url)

To explore what can happen in the area where the two concepts overlap or merge, each concept is firstly reviewed with regards to the literature discussed in the previous sections of this article.

**SAD T&L Reviewed**

SAD forms an integral part of systems development (Gorgone et al, 2002; Bajaj et al, 2004). Students need to understand a problem and subsequently design a solution (Albayrak, 2000). To accomplish this, the students need thorough SAD knowledge and skills, but also technical, business and management knowledge (Cybulski and Linden, 2000). Students do not necessarily have any systems development background (Suscheck and Huff, 2007).

There are various techniques, structured and object oriented (Gorgone et al, 2002), that can be used to come up with an undefined answer, but there does not exist a defined process to arrive at the best solution (Cybulski and Linden, 2000). Moreover, there is no single right answer or solution. It is therefore difficult to teach (Cybulski and Linden, 2000) and to learn (Smit, 2012). Students must be able to internalise concepts and processes, and develop their own thinking abilities. Students should be equipped with creative thinking skills, while outcome thinking, group process and communications are emphasised (Olfman and Bolston, 1992). SAD is applied in changing environments with changing requirements. SAD is therefore best learnt through application and experience seeing that this is how practising systems analysts improve their skills (Suscheck and Huff, 2007).

Common teaching methods currently used include case studies, simulations, real life examples, projects, and teamwork (Cybulski and Linden, 2000). However, the challenge with case studies is often that the students need in-depth understanding of the case or environment of application (Suscheck and Huff, 2007).
ATL Reviewed

ATL draws on the agile systems development methodology. The ATL approach responds to what is happening in, around and beyond the classroom (Berry, 2012). It encourages T&L that reacts to change by way of adaptation, rather than adhering to a predetermined plan (Berry, 2012). ATL encourages independent study with a view to lifelong learning, and makes use of iterations with variations (Chun, 2004). It proposes using immediate objectives, rather than a pre-planned sequence to a predetermined destination (Berry, 2012). The value of feedback is emphasised, and opportunity is provided for students to share their newly acquired knowledge (Chun, 2004). ATL adjusts and adapts in response to feedback and results obtained from monitoring the students. While it is important to provide students with frequent feedback, it is also important to create opportunities for receiving feedback from students (Chun, 2004).

ATL values working knowledge and suggests that opportunities for students to practice coursework is important. It involves constantly monitoring the progress of smaller units of work and encourages learner-centred and flexible teaching. The focus should be on the teachers, learners, and their interactions, rather than the teaching approach (Chun, 2004). ATL emphasises communication and collaboration, and proposes that students’ current knowledge can be used as a starting point for T&L (Berry, 2012).

Agile SAD Teaching and Learning

This section attempts to understand how ATL can be conducive to SAD T&L. The main connecting idea of the two concepts is the agile methodology that is taught as part of the SAD curriculum (Bajaj et al, 2004). ATL draws on the advantages that the agile methodology holds for the education setting (Davey and Parker, 2010). This paper attempts to apply the attractive qualities of the agile methodology to the challenges of teaching and learning SAD.

One of the challenges of SAD is the fact that knowledge from other fields is required (Cybulski and Linden, 2000) to understand a problem, in order to design a suitable solution (Alhayrak, 2009). ATL provides a possible solution in that it encourages independent study to produce lifelong learners. Independent study can be encouraged by providing further reading and study opportunities to improve the students’ knowledge about SAD and other knowledge areas. The students should experience the benefits of their independent learning by applying it to a SAD case study.

Various methodologies, agile and traditional (Bajaj et al, 2004), and techniques, structured as well as object oriented (Gorgone et al, 2002), can be used to perform SAD, which adds to the complexity of SAD T&L. When both methodologies are taught, the comparative advantages must be explained, as well as the fact that the two methodologies do not always contradict one another. The same applies to teaching both structured and object oriented techniques. Using iterations with variations of ATL could possibly address this challenge. The principle of iteration can also be used by applying the same methodology/technique repetitively to various problems. Likewise different methodologies/techniques can be applied to a similar case study through iterations to illustrate the comparative advantages. ATL values working knowledge (Chun, 2004), which readily agrees with the fact that SAD is best learned through application and experience (Suscheck & Huff, 2007). Using iterations with variation may provide more opportunities for students to apply their newly acquired knowledge to various cases.

ATL also encourages T&L that responds to changes and adapts when necessary (Chun, 2004; Berry, 2012). This principle can be applied to SAD T&L by accepting the undefined process of SAD and responding to the different ways students may apply the techniques/methodologies to arrive at various solutions. The marking of the solutions should then also be flexible and respond accordingly to different sets of students and their solutions.

Accepting that there is not one ultimate solution complicates SAD T&L. ATL recommend using immediate objectives, rather than a pre-planned sequence of steps to reach a predetermined plan (Berry, 2012). Instead of focusing on the undefined destination/solution or how to reach it, more immediate objectives could be used to define the process (Berry, 2012). An immediate objective for SAD T&L could be to create a case study purely for the sake of applying the new SAD technique/methodology, rather than creating great solutions. Only when the student feels more confident to apply the technique/methodology, create a case study to evaluate the actual solution. Repeated application of a technique/methodology should eventually improve the students’ solutions.

To increase the focus on communication (Olffman and Bolstron, 1992) rather than negotiation (Chun, 2004), offer students the opportunity to share their solutions with the rest of the class. Not only does this afford them the chance to participate in the teaching of SAD, but it also exposes them to the different possible solutions. SAD is not exact and precise (Smit, 2012), but ATL lets students share their newly acquired knowledge (Chun, 2004). Combined, this can potentially be a great way to confirm that there are various correct solutions to a problem while realising some benefits of peer instruction. It also provides an opportunity for monitoring student progress, and providing and receiving feedback.
Such sharing sessions could become crucial to SAD T&L, as it is critical for students to understand when to use which technique/methodology. When the students share their solutions, they can critique each other’s solutions as well as evaluate their own solutions against the others presented. Through this kind of feedback students can learn how other students have arrived at solutions, and they can share the challenges they have encountered.

ATL focuses on the teachers, learners, and their interactions with each other, rather than the teaching approach (Chun, 2004). SAD students must learn to internalise concepts and processes (Suscheck and Huff, 2007). Students’ sharing solutions and their learning experiences are learner-centred activities which encourage students to develop their own creative thinking abilities, which is an important skill for analysts and designers (Olfman and Bolstron, 1992).

The agile methodology accepts and reacts to changes, and takes an “inspect-and-adapt” approach (agilemethodology.org, 2008). SAD courses need to teach students to deal with changing systems requirements in changing environments. ATL uses frequent monitoring of smaller work packages to establish how and when to change and adapt one’s teaching (Chun, 2004). Through monitoring smaller units of work and giving as well as receiving frequent feedback, SAD teachers can determine the progress of their students, and direct or redirect their learning when necessary. In this manner you create a learning environment where the teaching can change as required. It provides the opportunity for students to become aware of, and accustomed and responsive to change. For a more direct approach, one could also change the requirements of a case study, thereby forcing students to adjust or redesign their solutions.

Case studies and projects, combined with teamwork, are common approaches employed to teach SAD (Cybulski and Linden, 2000). While the case studies and projects are used for students to apply and practice the techniques in SAD, the teamwork emphasises the importance of communication and collaboration, which are valued by ATL (Chun, 2004). A challenge of using case studies in SAD T&L is that students usually require an in-depth understanding of the case (Suscheck and Huff, 2007). ATL suggests that the students’ current knowledge can be used as a starting point (Berry, 2012). The students’ current knowledge could therefore be used to as a point of departure for a lesson or case study. When the case study is based on the students’ current knowledge, it at least partially attempts to address the challenge of the students not having an in-depth knowledge of the case.

From this discussion, one can construe that ATL can be conducive to the challenges of SAD T&L. What remains though is to explain practically how this can be achieved.

THE PROPOSED AGILE SAD TEACHING AND LEARNING APPROACH

To further understand what the practical implications are of adopting the ATL approach for SAD T&L, the research attempts to develop an agile SAD T&L approach. This is done by refining the discussion in the previous section into practical T&L processes with activities. The flow of the process is based on the ATLM developed by Chun (2004) with its iterative Teaching and Learning cycles. The process described below should similarly be reiterated with the necessary variations.

The approach starts off with establishing what students already know by asking about their demographics, previous education, work experience or systems development experience and knowledge. By using the students’ current knowledge as a basis, create a lesson to explain new concepts of a SAD technique/methodology. Provide the students with some additional reading about the technique/methodology or the application thereof, to encourage independent study. Follow up with a class discussion or assessment to retrieve feedback from the students. Use the feedback gathered to determine if the students understand the new concepts of the technique/methodology and how the reading aided their understanding.

To establish the new concepts, develop an exercise or opportunity for students to practice the newly acquired concepts of the technique. Again use students’ current knowledge and previous experience, but this time to set up a case study. The case study should follow on the lesson and the extra reading. If the students are involved in creating the case study, then they have at least some understanding of the case. Keep in mind that you should set up the exercise to meet an immediate objective, and that this single exercise will not determine whether your students can become good system analysts and designers. When the SAD technique/methodology is used for the first time, then the objective of the case study is to provide the students with an application exercise to test their understanding of the technique/methodology. Only when the technique/methodology has been applied before, can the objective of the case study be to evaluate the solutions.

For the first application of a technique/methodology, basic steps or guidelines for application can be provided, but not with too many specific details. When the technique/methodology has been applied before, provide guidelines based on the feedback or results from the previous iteration. The students should have some space to develop their own way of applying the technique/methodology to the case. This will promote internalising the process and concepts of SAD.
These exercises can be done individually as long as it is followed up with a team effort. This encourages collaboration and communication amongst the students and also facilitates some peer instruction. The solutions to the case study can then be shared in class. Again this creates an opportunity for communication and interaction amongst the students and with the teacher. The teacher should guide discussions and debates around the suggested solutions. Through this the students will learn that there are various ways to analyse a problem and to design a solution. The discussions should focus on critically evaluating the application of the technique/methodology or the solutions. This depends on the objective of the case study. The discussion should be guided towards establishing an understanding of how one decides when a solution can be considered good or not. Provide an opportunity for students to reflect on their own work and adjust their thinking accordingly, either by writing a lessons learned report, or having a discussion about the challenges they incurred, and how they could have handled it differently. Encourage creative thinking.

To monitor the progress of the students in terms of their ability to evaluate how good a solution is, the students should be allowed to give feedback by for example voting for the best solution. The teacher should then consider the progress of the students and change and adapt the teaching or learning experience as deemed necessary.

The essence of the T&L processes described is shown in the diagram below.

![Figure 2: The Agile SAD T&L Approach](image-url)
The T&L processes happen simultaneously and are inseparable. The whole process should be reiterated as many times as considered necessary, with variations. Variations could include solving the same or a similar case with a new technique/methodology or applying the same technique/methodology to more than one case study. The requirements in the case study could also be changed during the process to teach students to adapt. Another option is changing the team members to encourage more interaction amongst students and increase exposure to different ways of thinking and doing. Instead of voting, the students can mark each other’s work, or the teacher can explain how the solution is evaluated. Of course any other variation a teacher deems necessary to improve the teaching or learning of SAD can be incorporated, since the approach is meant to be agile.

CONCLUSION

The agile SAD T&L proposed in this paper uses the attractive qualities of ATL to provide possible solutions to the challenges of SAD T&L. The approach suggests using iterations with variation to address the challenge of teaching various techniques/methodologies in a SAD course. ATL focuses on people and their interactions, therefore agile SAD T&L proposes that the teaching should respond to the students’ needs. Frequent feedback sessions provide the opportunity for students to share their newly acquired knowledge and learn from each other. Using independent study is recommended to address the challenge of students needing technical, business and other knowledge. Another suggestion is tapping into the students’ current knowledge to ensure they have some knowledge of the case studies used. Instead of depending on predetermined steps to reach predetermined destinations with predetermined lessons, ATL encourages flexible, adaptable teaching with immediate objectives, responding to the feedback of the students, and making necessary changes.

The basic principle of the agile SAD T&L approach is to be flexible and adaptable. The approach should thus adjust to the T&L styles, and not the other way around. Agile SAD T&L does not expect teachers and students to adapt to the new approach, but rather to apply the approach as it suits them best.

The suggested approach seems promising, but is purely based on the literature reviewed, and has not been fully tested. Further research calls for testing the approach, reiterating the approach with variations, monitoring the progress and adapting the approach as needed to confirm its legitimacy. This is currently done by the researcher as a SAD teacher within the typical time constraints of a course. The researcher hopes to inspire other researchers to conduct further research in this area and thereby mature and contribute to agile SAD T&L.

REFERENCES