13 Common Agile Issues Identified and Addressed

By Brian Vanderjack, PMP®, MBA
Agile Project Manager, Scrum Master and Agile Trainer – AT&T
and
Lisa Clements, Agile Coach

Introduction

I work for AT&T as a project manager. In my area we just finished our first IT Agile-Scrum (to be referred to here as simply Agile) Project. While leading this effort some Agile procedural questions came to the surface that causing team confusion. Some questions I was able to address on my own after giving them some thought. Others I looked to the AT&T Professional Agile Coach, Lisa Clements, for guidance. What is here represents a blend of what to look out for and solutions that have worked for me. Lisa and I have invested a lot of time into this material and hope it saves you hours of frustration by illuminating some sticky points of using Agile. And more importantly, I hope it assists you in implementing Agile with as few issues as possible so you are more likely to adopt this valuable methodology for the long term. Please note; the reader is assumed to have basic understanding of Agile and its related terms. Also, this article does not, in any way, represent AT&T's position.

Brian Vanderjack, PMP, MBA

#1 Question: I hear Agile only works only for small web based projects.

Answer: Agile works for large as well as small projects. There is more work needed and/or investment to keep large Agile projects on track as opposed to small ones. If a team is tackling a large project:

- They will need a software tool to keep track of all of it. It is possible to get by using spread sheets, but the amount of overhead could be tremendous.
- Significant up-front communication and planning along with continuous communication are needed as product backlog items are added.
- If multiple projects are being run concurrently, as part of a program, sync points for testing must be determined well before testing actually starts. That is, what each project needs (that is test media or processes) to do testing is to be determined, and schedules are to be set up that allow systems to have access to the test media they need at the time they need it.

A key point in Agile is the overall risk to the business, in terms of not meeting the Product Owner's expectations on a very large project or program, can be reduced through smaller, more frequent deployments.

#2 Question: In an Agile environment, what is the difference between a Project Manager and a Scrum Master?

Answer: A Project Manager is charged with driving the project to conclusion with an eye toward keeping the project timeline and money boxed. The Project Manager tends to support multiple scrum teams and projects at one time. A Scrum Master is a servant-leader whose focus is to ensure the Scrum Team meets its commitments for each Sprint and ensures constant communication with the Project Manager.

Another way to answer this is hidden in the Agile Manifesto (Agilemanifesto.org, 2001). These four statements are from the manifesto:

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan

Project Managers tend to favor those items on the right, whereas Scrum Masters tend to favor those items on the left. The Project Manager actually stays in contact with leadership and others as appropriate and has ultimate responsibility that the budget is met. The Scrum Master on the other hand protects their team from outside influences and works through the Project Manager to resolve issues that can't be resolved at the Scrum Team level.

#3 Question: Our team is having difficulty identifying a knowledgeable Product Owner. It appears the knowledge we need is not in a centralized location. We need a single point of contact as our Product Owner, what should we do?

Answer: This is not an easy solution. If knowledge is not centralized, then the various Product Owners (or business representatives) need to elect one Product Owner that can gather requirement and clarifications from the other Product Owners and act as the one point of contact for the Project Manager and Scrum Master. They can also appoint a Product Owner Delegate who is typically an Analyst to act on their behalf. This delegate attends all the ceremonies in place of the Product Owner and has authority on their behalf to prioritize, create, and accept User Stories.

#4 Question: What projects work best with Agile?

Answer: Here is a short list (not-all-inclusive) of the key success factors of a project using Agile:

1) Executive level support. It is best to have the support of leadership going into a project to maintain positive stakeholder support. For example when a Product Owner eventually complains to leadership that the team will not agree to a scope change that leadership understands the process before hand.
2) A supportive product owner or product owner delegate. It takes time and commitment for the Product Owner to perform their role. They tell the team what the business wants, prioritize those wants, provide real-time feedback about user stories, and actually test the completed work package at the end of each Sprint. As a side note, a great candidate for the delegate role is an Analyst.

3) Separable work products. That is, since Sprints tend to be around 2 weeks long, being able to break work products up into 2 week chunks is essential. With practice most Scrum Teams find there are few things that can't truly be broken down into steps that can be done in a 2 week period.

4) Minimize the involvement of waterfall projects in your program. If there are waterfall efforts that will need to be integrate with the Agile project, readily identify sync up points with those waterfall projects where you can have meaningful system tests along with communication plans for status, impacts of new backlog items and common and shared scope change requests.
5) You can use Agile on projects that do not meet the above list of criteria. Just realize that for each criteria of the optimal Agile project that is not met, more risk will be added to the project and the more important efficient and effective communication will be throughout the project.

#5 Question: How are you supposed to hold a demo if you have nothing working at the end of a particular Sprint to show the Product Owner?

Answer: People learn best by seeing, hearing, and touching (experiential). Identify a method that you can use to share what you have done with the Product Owner that maximizes the use of the three methods people learn by. For example:

- Build a flow chart for visual people
- Provide a verbal walk through for audio learners
- Use a simulator to make it as real as possible

Clearly, having actual working software, and something to demonstrate, is a superior to alternative methods of demonstrating completion at the end of a Sprint. Remember, if you have nothing working yet that you can share with the Product Owner the more of the three learning methods you can incorporate during a demo the better.

#6 Question: What if a Sprint can't deliver on time?

Answer: Identify which user stories can complete on time and which user stories have the highest priority. Use that to identify what you will push out to the next Sprint. If it is the last Sprint that you are working on when you realize you've fallen behind, you will need to have the same conversations you have when a waterfall project cannot complete on time. The project may need to move out or deploy what is completed and accepted. What is not completed could be added to the backlog of a future project. If a team is doing a Daily Scrum meeting and bringing up impediments, as they should, it would be extremely rare for a team to discover in the last Sprint that they can't complete the project on time.

As a part of the retrospect, you will need to figure out what went wrong and fix it. That is, do a root-cause analysis like Fishbone, or "5 Whys." If you don't, the problem will continue to haunt your ability to complete even if you get an extension. Each Sprint should give you the opportunity to change something, to fix what isn't working well and adjust future efforts until success is realized.

On the bright side, if you get gummed up on the last Sprint, the items, by definition, will be low value add items/User Stories. You may be able to jettison work from the project, and still keep the vast majority of the value of the project intact.

#7 Question: Is it easy to start using Agile?

Answer: Agile requires a change in mind set and that is rarely easy. A good rule of thumb for your first project is to take the work that you believe you can complete in 2 weeks and only give your team half of that work the first Sprint. Bruce Tuckman's famous forming, storming and norming will need to take place, and that will eat up a considerable about of time during your first Sprint. However, one value of Agile is as Scrum Teams stick together for a very long time the start-up thrashing is minimized. Realize part of Bruce's model is that you will be kicked back into previous stages, as change occurs or personalities clash. But, at least the team culture is set up and that is the most time consuming part.

Part of norming is just learning how to do Agile which takes time, too. For example, you will need to integrate with the incumbent business model to get stuff done. Also, having to buy a tool to help you manage your Agile project, internal procurement processes and the learning curve to use the new tool may eat up valuable time. Remember that one of the benefits of Agile is that it's okay to fail, but do it fast and correct it as you go. Use your first project to establish your teams, processes, and get comfortable with the process.

#8 Question: How do you manage waterfall and Agile projects when they take place concurrently in the same program? To explain, Agile demos and testing takes place every 2 weeks, and waterfall may not be testing for months after the program starts.

Answer: The Project Manager and Scrum Masters need to work closely with the other applications during project and release planning to set up points where various forms of testing can take place. The Agile teams may need to do stubbed out testing or test with mock interfaces, data, processes, or similar until the waterfall test environment is ready to do a basic test. The key is to identify points where the waterfall team will be ready to do some basic "can we talk to each other" testing then incorporate user stories or time into your velocity during the full blown end-to-end or integrated test.

#9 Question: "Lessons Learned" tend to be a joke in waterfall development methodology. Lists of issues get created, but rarely does an improvement get enacted. How are Retrospectives in Agile different?

Answer: One special attribute about Agile teams is they stay together. Assuming the team has a written process that they follow, the process can be shaped as more is learned about employing the Agile methodology. It is up to the Scrum Master to make sure the process is updated and new key-learnings are employed in future Sprints. Retrospectives (the name for Lessons Learned in Agile-land) occur at the end of each Sprint and discuss what worked and what didn't.

Part of that meeting is to pick one or two things that didn't work and come up something to try in the next Sprint to fix it. One critical success factor is the Scrum Master needs to de-couple the problem found in the retrospect from the person involved in the mistake. That is, don't say something like "Sue's mess up" or Sue will never participate in Retrospectives again. The reality is that it may be difficult to channel Scrum Team discovered retrospects into the organizational consciousness. But at least when the learning specifically impacts the Scrum Team they can improve through use of knowledge captured during the retrospect.

The three big issues that traditional waterfall project teams face with a Lessons Learned process are:

1) When the project completes they disband and there is little chance to have that team carry forward what it learned.
2) Creating process improvements as a reaction to lessons learned takes an actual skill set which may not be resident within the project team.
3) Allocating time and money to fix processes will likely require funding. The team may not have access funding that is not directly tied to immediate cash value. Fixing stuff just may not be a huge organizational priority.

#10 Question: What if executive support for moving to Agile is not present?

Answer: Generally executives don't care "how" something is done as long as metrics are met and the time, cost, and quality expectations are, too. There will be bumps when you start using Agile, as well as expenses like hiring Agile coaches and buying software that supports Agile. It is important to educate leadership that the first one or two

projects you will see no extra value to using Agile. Sharing a brief overview of Agile with leadership and describing the expected benefits can help build an understanding. However, once improvements are realized, the Scrum Master and Project Manager should share and promote their team's efforts. The Agile Burn Down Chart is a great visual where even a busy executive can understand the health of the project at a glance. Giving access to this tool up front can also help executives understand things will be different with Agile, but in a good way.

#11 Question: How can a Scrum Master deal with a client who changes their mind after the product is produced?

Answer: This brings up two challenges. First Agile User Stories should include acceptance criteria and the definition of 100% complete and delivered (often referred to as "done done"). This should all be spelled out by the client long before they see results in a demo. That way, shockers like are minimized. Some implementations of a User Story do cause the Product Owner to change their mind e.g. "I could have 2 more fields on this screen and not need that second one I defined." If this happens, they can add a new User Story to the backlog and prioritize it as they see fit. This may result in other User Stories being cancelled as they are no longer needed. It may also require the conversation with the Project Manager, as in water fall.

When scope changes, a determination of what will have to fall off the list or how much additional funding will be required and any time slips will always be needed. If a client changes their mind mid-Sprint, then the story should be stopped and a new one added to the backlog and prioritized.

The second challenge is when the Product Owner is not engaged. Product Owner should be so involved, that they know what is happening, long before the demo. There should never be a surprise at demo time.

#12 Question: How do you deal with a client who thinks they have unlimited right to add and change?

Answer: It is my experience that all projects are a lot like working on a fixed bid contract. To explain, you do your initial estimates and then get funding. If there is a need to play with the initial budget, you have to go back and ask for more money. If you ask for too much money, bad things start to happen like:

1) Project Manager/Scrum Master replacement
2) Project cancelation
3) Financial audits
4) Heavy handed executive "help."

What the Project Manager and Scrum Master need to do is set expectations up front and share constant reminders with the Product Owner that the Product Owner (or

someone similar) is paying for the project. They need to be taught and reminded that as User Stories are added, they contend with every other user story for funding and time. The Project Manager also needs to be doing a financial comparison of budgeted amounts to actual amounts spent at each Sprint. These results are to be shared with the Product Owner during Grooming so that they have a good understanding of how much has been spent and how much budget is remaining. This helps them understand how much latitude they have in terms of future scope changes.

By understanding this and clearly communicating how scope changes are driving low priority User Stories out of the scope of the project, it should rein in this behavior. But remember, that is what Agile is about: Change. This is where doing small amounts of work in short timeframes pays off. The client can change their mind and make the decision each Sprint as to what is really important.

#13 Question: How do I identify what user stories to slot into a Sprint?

Answer: First the Product Owner assigns a priority to each User Story. Then the items in the product backlog are estimated using "Story Points." Story Points are an identification of relative effort of a User Story. It is a very quick way to get a handle on how much work a particular user story will be and use it for slotting work into a particular Sprint. The maximum about of Story Points, in total, that can be worked during a Sprint is referred to as Velocity.

Here is a typical example of the potential Story Points that can get assigned during planning: 1,2,3,5,8,13, & 21 (number crunchers call this the Fibonacci sequence). Once the team decides what numbers will be used and what the numbers mean, each team member assigns a Story Point number to each User Story. If a User Story has more than one Story Point number assigned to it by the team, the team members repeat this voting process until, with conversation and hidden voting, each team member eventually assigns the same Story Point value to a User Story. Physical playing cards can be used to do this, or there is a web site that can be used to do Story Point voting if team members are in remote locations (PlanningPoker.com).

When work is slotted into a Sprint, the highest priority user story gets slotted first. Each time a user story is added to a Sprint, the related Story Points are added into a counter. If the counter is near the agreed upon Velocity and the next User Story would go above the agreed upon Velocity, one of three things can happen:

1) The highest priority item can be decomposed into smaller user stories. One (or more) of these child user stories can be slotted instead of the initial user story
2) The next highest priority item that can fit within the velocity is slotted.
3) The team can cheat and go over capacity for the Sprint (as dictated by Velocity).

However if this is done, problems with timely completion of the Sprint can be expected.

One thing to keep in mind when planning velocity is to look at the backlog, as a whole, for the entire Scrum Team, not just the project. It as it is rare for a Scrum Team to work on just one project at a time, so multiple demands on a team must be considered. The Scrum Master needs to take into consideration the percentage of time to be spent on each project in the Sprint and make sure the Story Points are allocated accordingly to each project.

References

Agile Manifesto. (2001). The Agile Manifesto. *The Agile Alliance*, retrieved from the AgileManifesto.org

About the Authors

Brian Vanderjack

AT&T Mobility/Home Solutions

Illinois, USA

Brian is blessed to have a wonderful wife, named Terri, and two sons Matt & Marc. Matt, just finished his criminal justice degree at Western Illinois University®. Matt composes music, on his PC, in his free time. Marc, our English major at Illinois State University®, is currently away studying in England. We also have a long haired Wheaton Terrier (Finley) who regularly awakens us at 4 AM because he is board and wants to play outside. Brian's passion is providing people the skills they need to be better at delivering what their employers need. And, by being even more valuable, the expectation is they will provide longer service to their home organization.

Brian is honored to have been: appointed Scum Master at AT&T®* for the award winning MyCSP App, point of sale information support application (top-ten design award, *worldwide*, by the Neilson Norman Group® 2013), MBA with distinction (DePaul®), faculty of the year (University of Phoenix®), Excellence in Project Management Award (AT&T®), and often published (in PMWorldToday® & PMWorldJournal®).

Brian has taught PMP® certification classes for DeVry®/Becker® and AT&T®, he has also taught Agile at AT&T® and for the PMI® Chicago land chapter (Professional Development Day). He also teaches Project Management for the University of California – Irvine®, and University of Phoenix®. Brian has taught or spoken at many other venues like Microsoft® and Northwestern Kellogg Graduate School of Business®. He has provided seminars & keynotes on leadership at AT&T®, Toastmasters®, several PMI® chapters, and Abbott Labs®.

For further background about how Brian continues to drive toward fulfilling his mission, find his 44 written recommendation and hundreds of "referrals" on LinkedIn at www.linkedin.com/in/brianvanderjack/ . Brian offers face-to-face PMP certification training on Saturdays. If your organization, or PMI® Chapter, is interested, reach out to Brian at BrianVanderjack@Gmail.com (no-pass = tuition refund, but conditions apply).

**Note: For the above, endorsement by identified organizations is not implied or to be inferred, their trademarks and brands remain their own. As Brian is full time employee of AT&T, the following is required per AT&T corporate policy: Information provided in this BIO and the related article in no way represents AT&T, nor its processes. Due to a non-compete agreement, services can't be provided to organizations that compete with AT&T.*

Lisa Clements

Lisa Clements earned her BS in MIS from the University of Idaho, where she also studied computer science. She then began her career at Digital Systems Int'l as a database programmer. She was soon recruited by AT&T where she took on a management role, managing software development, architecture, and project-management IT teams. She credits her success in these roles to the principle she instilled in each of her staff members: *Deliver exceptional value to AT&T's customers*. For her superior efforts, she was named one of AT&T's top 300 employees and thus awarded the coveted Circle of Excellence award. Her career then moved onto development in the Agile framework. She was selected to be a member of the effort to bring Agile into AT&T, and she worked on one of the first AT&T Agile development teams. She then moved to helping create, then implement a company-wide Agile coaching program as part of the Agile Circle of Excellence program. She also was awarded the Center of Excellence and the Customer Rules awards, from AT&T, for delivering excellent service to the company.