

조심해! 애자일적용의 다섯가지 Anti-patterns



Robert Galen - February 2, 2015

Bob Galen 은 애자일 방법론 코칭&트레이닝 컨설턴시를 근간으로 한 RGCG, L.L.C. a Cary, N.C.의 President 이자 Principal consultant 입니다. 그는 Agile 커뮤니티에서 왕성하게 활동하며 풍부한 경험을 가진 Agile & Certified Scrum Coach (CSC)입니다. Agile 방법론과 관련된 모든 주제에 대해 정기적으로 집필을 하며 많은 이들을 가르치고 있습니다. Bob 은 agile 관련 두가지 책을 집필하였습니다. Agile product ownership 을 둘러싼 업무를 정의한 "Scrum Product Ownership", 그리고 mature agile patterns 에 초점을 둔 'Agile Reflections'입니다. bob@rgalen.com 으로 메일을 보내거나 <http://eepurl.com/IAxTD> 에서 메일링 리스트에 등록하시면 그와 연락이 가능합니다.

SHARE

Related Topics: [Scrum](#)

거의 매일, 저는 열정적이며 극단적으로 "애자일 방식으로 전환중인" 팀들과 직면합니다. 그들에게 있어 이는 삶 전체가 바뀌는 이벤트입니다. 그런 그들의 생각들은 다음과 같이 표현될 수 있습니다.

- 리더와 매니저 : 더 많은 물건을 팔기 위한 force multiplier(전력승수)로 여김
- 팀 : 상부의 관리 없이 즐기면서 소프트웨어를 만드는 방법으로 여김
- 프로젝트 매니저 : 파격적인 혁신과 그것에서 오는 '치명적인 두려움'으로 여김
- 그외 모두 : '참여하지않고 관망하거나 소극적으로 가담하는 것' 같음

그들에 대한 위의 표현들이 어느 정도는 진실성을 갖습니다. 저는 '여러분은 인생의흐름을 잘 타고 있습니다'라는 관점에 전적으로 동의합니다. 이런 생각과 감정들은 종종 저에게 질문들을 불러 일으킵니다. 애자일 코칭 경험에 따르면, 종종 애자일의 방법과 실행(Agile tactics and practices)에 대해 효과가 있는 것과 효과가 없는 것이 무엇인지 질문을 받고는 합니다.

많은 이들은 '묘책'과 같은 답을 찾고 있습니다. 예를 들어, 저에게 플래닝 포커(Planning Poker)를 사용하여 어떻게 평가하는지, 혹은 전력질주를 위한 적절한 범위는 무엇인지 묻는 것처럼 말이죠. 저는 애자일의 복잡함과 상황적인 속성 때문에 직선적인 답변을 주는 것에 대해 자주 고심하게 됩니다. 딱 떨어지듯 명확한 하나의 답은 없습니다. 각 팀의 유용성은 변할 것이고, 그들은 각자의 것들을 폭넓게 고려해야만 합니다. 그리고 그런 애매모호함이 종종 저의 고객과 학생들을 좌절시키곤 합니다. 그들중 많은 이들이 그들 스스로의 것으로써 깨닫고 발견하는 것보다, 단지 어떻게 해야 할지 간단히 묻기를 원합니다.

(하나의 답은 없다고) 그렇게 말은 하지만, 애자일을 잘 수행하기 위한 어떤 생성 행동과 법칙은 있습니다. 본문에서는 애자일 여정에 '무엇을 하지말아야하는지'에 대한 것을 공유하기 위하여 공통의 *anti-patterns 집합을 탐구하고자 합니다. 제 바램은 이것이 당신이 (애자일로의 전환을 위한 것에) 좀 더 나은 실행과 방법(practices and tactics)을 위한 벤치마크로써 도움이 되는 것입니다.

(그러나 다시말하지만, 이것들이 당신의 환경에 맞도록 잘 생각하고 적용해야한다는 것을 명심하십시오.)

** Anti-patterns : 패턴과 반대로 프로그래밍을 하는 과정에서 프로그래머들이 흔히 범하기 쉬운 바람직하지 않은 방법들과 그로 인한 폐해를 의미 (번역자 주)

시기적으로 빠른 애자일 적용 : 다섯가지 Anti-Patterns

1. 개발자 주도의 애자일 : 전체팀의 접근보다, 개발자들은 (애자일) 채택의 대부분을 주도합니다. 저는 몇 년전 120 스크럼 팀을 가진 한 고객을 코치했었습니다. 비록 그들이 조직내에 300 명 이상의 테스터들을 갖고 있었다고 하여도, 스크럼마스터들의 모두는 개발자들이였습니다. 당신 역시 이것이 평가와 계획에 중점을 둔 것이라는 것을 알 수 있을 것입니다. 개발자들이 중요하다고는 하나, 애자일리티는 사실 서로를 존중하고 감사해야하는 Team sport (단체경기) 입니다. 팀과 결과물 모든 측면에서 모든 기능을 포함해야합니다.

** 스크럼(Scrum) : Project 를 위한 상호, 점진적인 개발방법론이며, 애자일 소프트웨어 공학 중의 하나이다. 스크럼(Scrum)은 소프트웨어 개발 Project 를 위하여 고안되었지만, 소프트웨어 유지보수 팀이나 일반적인 Project/Program 관리에도 적용 가능하다. (번역자 주)

이 패턴의 일부분은 기능(function)별로 프로세스를 평가하고 있습니다. 예를 들어, 개발자들이 사용자 스토리를 제작하는 과정에서 자신들의 부분을 작업하고 나서, 그것을 최종완성하기 위해 팀의 다른 사람들에게 넘기고, 그 후 그들은 더 많은 코딩 작업으로 넘어갑니다. 그래서 측정은 기능적입니다. (프론트 엔드 개발, 백엔드 개발, 테스트, 분석 등). 이것은 팀 delivery 를 덜 최적화시킵니다. 대신, 사용자 스토리를 함께 가능한 빨리 완성하기위해 모두가 집중하기 훨씬 좋습니다.

Healthy Pattern : 애자일은 포괄적이고, 공명하며 협력적이고 열려있는 '단체경기'임을 기억하십시오.

2. 분산된 팀이라는 핑계: 저는 아마도 일주일에 적어도 두세번 이 같은 질문들을 받습니다. 질문들은 보통 이렇습니다 : 우리는 분산된 팀이고, 애자일의 협업적인 측면의 많은 부분이 우리를 힘들게 합니다. 페어링, 코드리뷰들, 스탠드업, 함께 계획하고 단기간의 리뷰(스프린트 리뷰) 등의 일들은 불가능합니다. 이런 어려운 일을 건너뛸 수 없을까요? 저는 사람들이 제 대답과 같이 생각한다고 생각하진 않지만, 제 대답은 항상, '아니오, 그럴수 없습니다.' 입니다.

분산된 팀으로써, 당신은 설사 그들이 힘들어하더라도 확고한 애자일 팀 협력 원칙을 분명히 해야합니다. 협업을 보다 쉽게 할 수 있는 기술(tools, video, seating, flying folks around 등)을 위한 당신의 리더쉽이 필요하되, 협업을 최소화하기 위한 구실로써 분산된 애자일리티를 사용하지는 마시기 바랍니다.

Healthy Pattern : 분산된 팀이 협업실행에 전념하고, 함께 작업하는 팀으로써 가능한 많이 움직이도록하여 혁신적으로 그들의 능력을 향상시켜라.

3. 인터럽트 주도적 작업: Tom DeMarco 가 1997 년에 공동집필한 Peopleware 에서 그는 정보취급자(지식근로자)들에게 최소한의 중단과 멀티태스킹에 따른 집중의 힘에 대한 사례(practices)를 만들었습니다. 저는 멀티태스킹이 당연히 빠를것이라고 느끼는 것과는 반대로 결국에는 그렇지않게 끝난다는 것을 모두가 알것이라고 생각합니다. 그럼에도 불구하고, 우리는 여전히 선택을 해야할 때 매번 우리 스스로 자제력을 잃곤 합니다. 멀티태스킹 비용을 위한 최소화, 요약 그리고 중요정보공유는 좋은 애자일 팀들의 특징입니다. 그리고 절제력은 당신의 조직에 대한 외적인 부분과 팀의 내적인 부분, 둘 모두에게서 나옵니다. Heroic firefighting 은 중독될 있기에 각각을 이런 혼돈의 형태로부터 떼어낼 필요가 있습니다.

Healthy Pattern : 집중은 일을 깔끔하게 처리하기 위해 아주 중요합니다. 코어 애자일 테넌트를 강화하기 위해 모든 톤에서의 중단과 업무전환이 일어나는 것을 줄여보세요.

4. 리더가 주도권을 잃은 팀: 아주 일반적인 anti-pattern 은 팀들이 원하는데로 무엇이든 하도록 내버려두어 조직은 혼란스런 자기 결정권(self-direction)을 가지는 경우입니다. 원리는 애자일 팀이 조직의 규칙이나 통제없이 본래 스스로 모든것을 찾아내도록 하는 것입니다. 저는 self-direction 을 지지하고 있었던 사실 때문에 어떠한 가이드도 제공하지 않는 팀을 코치한적이 있습니다. 심지어 제가 요청을 했을때조차도. 그것은 터무니없는 일이었습니다.

견고한 애자일 팀에서는 리더쉽이 필요합니다. 통제는 그들이 "종료"의 분명한 정의, backlog 작업 그리고 목표설정과 계획에 대해 표현할 때 팀을 강하게 만듭니다. 그리고 팀은 그들이 정의했던 애자일 기본규칙을 믿고 존중함과 동시에 견고한 행동방식 기준(표준)을 따라야만 합니다.

Healthy Pattern : Self-directed 팀에는 지시가 필요하다는 것을 깨달아야합니다. 그것을 적용할 효과적인 통제와 가이드라인이 있습니다.

** Inmates are Running the Asylum : 전문분야와 기초에 박식한 팀원들이 모든 것을 결정하며 매니저가 그에 따라가는 상황을 표현하기 위해 사용되는 문구입니다. 영어단어상으로는 '수감자가 탈옥하려고 한다'는 의미

5. 피상적인(무의미한)애자일: Scrum-But(Scrum 의 역기능)이라고도 부릅니다. 팀이 아주 작고 간단한 Agile Practice 을 말으면, 피상적인 적용으로 인해 전체의 중요성이나 ROI(Return of Investment, 투자이익률)을 놓치게 됩니다. 예를 들어, 팀들이 Scrum 으로 움직이고 있을 때 대부분은 stand-ups, backlogs 그리고 iterations 을 채택합니다. 하지만 더 많은 통제가 필요한 '어려운 일들'의 일부는(종료의 정의 설정이나 ScrumMaster 집중 등) 자주 간과되곤 합니다.

일반적으로, 나쁜 사례는 -모든것이 어느정도 완성되었을 때- 추후에 적용된다는 것입니다. 그땐 조직내에 추가 비용의 여유가 없습니다. 삶의 모든것들이 그렇듯 진짜 변화와 효과를 위해서는 힘든일과 투자가 필요합니다. 저는 "지불한 만큼 얻는다"고 이 팀들에 말해주곤 합니다.

Healthy Pattern : 그것이 어렵고 힘든일일지라도, 애자일 방법과 실행의 견고한 조치(정책)을 채택하십시오. 창조적인 모델에 투자하고, 완성도를 쌓으며, 그것이 변화하기까지를 경험하십시오.

마치며,

저는 이 패턴들에 대해 불평을 하는것도, 그런 패턴들이 보이는 조직, 매니저 혹은 팀들을 비난하고 있는 것도 아닙니다. 저는 여러분들이 그런 anti-pattern 들을 피하기 위해, 또 쉽게 그 패턴들을 느낄수 있도록 간단히 나열하였습니다. 과거에 행했던 이런 것들을 확인하는데 도움이 되고, 그런 당신의 anti-pattern 들을 피함과 동시에 유익한 패턴들로 향할 수 있도록 말이죠.

보다시피, 일반적인 견해와는 다르게 애자일은 결코 쉽지 않습니다. 저는 '올바른'것(패턴)을 얻기 위해서는 통제, 연구, 열린마음 그리고 힘든 과정이 필요하다고 믿습니다. 또한 이는 최종 결과물의 획득보다, 더 많은 발견과 지속적인 발전의 여정이 될 것이라고 생각합니다.

아마 그것은 흡사 *Lost in Space* 의 어딘가의 (우주)정류장일지도 모릅니다. 그들은 지구의 집으로 가는 길을 알려주지 않기에, 당신 스스로 당신과 당신팀을 위한 진짜 세상으로 이끌어줄 건강한 애자일 적용으로 가는 길을 찾기를 바랍니다.)