

Note: This excerpt was taken from the book

72 Reasons Why Scrum Works

Part of

The Scrum Mega Pack

By

Paul VII

© 2012 by Paul VII

Published by

PASHUN PUBLISHING

A division of

Pashun Consulting Ltd.

PO Box 36033

London

England

www.pashunconsulting.com

72 Reasons Why Scrum Works.....	2
Introduction.....	6
10 reasons why AGILE PRINCIPLES work.....	7
Background and relation to Scrum.....	7
Reasons.....	7
5 reasons why THE SCRUM MASTER ROLE works.....	8
Background.....	8
Reasons.....	9
5 reason why THE PRODUCT OWNER ROLE works.....	9
Background.....	9
Reasons.....	10
5 reasons why THE DEVELOPMENT TEAM ROLE works.....	10
Background.....	10
Reasons.....	11

Introduction

If someone asked you "Why does anyone even bother using scrum? " how would you answer them? Since the year 2000, I have been asked this question many times, in many different ways, at different companies, by people in different roles.

Scrum is a very easy framework to understand but an extremely challenging one to implement. This is not usually because its practices require special aptitude for learning or even years of experience, but because of the emotional, ideological and other real world challenges that we all face in business.

In my experience as a scrum master, team lead and software engineer, the question "Why?" has been asked at every level of the business in one way or another. This is because scrum is a new way of thinking. It requires a change in what we do to deliver projects, and therefore any reasonable human being will want to understand why it is worth changing old habits. Answering these questions will help you to understand why the practices are so worth while on your projects.

Realising how important the answer to the question "Why?" is, I have condensed hours of research and experience into one easy guide. This can be used to recap, up-skill quickly and most importantly, answer questions for people who want to run their projects more efficiently but do not know why scrum is the right way to go.

10 reasons why AGILE PRINCIPLES work

Background and relation to Scrum

Before agile, the waterfall model was seen as the industry standard method of software development. Waterfall consists of upfront phases of Analysis, Design, Implementation, Testing and Maintenance. These steps proved difficult to embrace change and made businesses wait until all steps were complete before the product was released. Therefore, any return on investment was not made until the end. Agile is an alternative way of thinking. It is an umbrella term for a set of methods that share a number of principles such as working in iterations (an iteration is usually a period of one to four weeks) delivering working software early, and finding simple solutions to complex problems. Scrum is an agile framework. Other examples of agile methods are XP, DSDM, crystal and FDD.

Scrum consists of three agile principles: transparency, inspection and adaptation. Transparency means that the whole team trust each other and are truthful about whether tasks are completed or not. Inspection means that they assess their progress and adaptation means that they make changes for the better based on their inspection. The product owner builds a backlog of features. Each iteration (sprint), the team takes a subset of these features, works on them and delivers a potentially shippable product increment.

Reasons

1. Earlier return on investment: Working in short iterations means that return on investment is delivered early.
2. Responds to the market: The business can change requirements after each iteration and respond to changing market conditions.
3. Early testing: Everyone involved can see working

software in action before it goes live. This means early testing by the business and end user.

4. Product meets expectations: Agile explicitly brings the customer, end user and development team close together, so that what is being built can match the end user's expectations.

5. Product corrects expectations: If something cannot be achieved for launch or will not perform as expected, everyone concerned will know early on.

6. Strong, motivated teams: Trust is at the foundation of all agile methods. This motivates all involved and builds strong teams.

7. Highly efficient teams: Using agile principles means rapid communication. It also forms more efficient teams through co-location and co-ordination meetings. No technology can substitute this human interaction.

8. Working software early: There is nothing more important than seeing a piece of the finished product. Agile practices deliver this each iteration.

9. Teams get better with age: The concept of continuous improvement means that focus on technical excellence and good design is constant.

10. Simple but effective: Agile teams crack problems using the simplest method possible, saving themselves and the business time and money.

11. Accountable team more likely to deliver: Unlike other methods, the people doing the work are held responsible for delivery (as opposed to the project manager). This creates a more natural sense of urgency.

5 reasons why THE SCRUM MASTER ROLE works

Background

The scrum master is the resident scrum expert, coach and

mentor for all involved in the project. It is important for the scrum master to fully understand the rules of scrum and make sure that the framework is carried out as per these rules. The scrum master is also responsible for removing any impediments to the project's delivery.

Reasons

1. **Dedicated bulldozer:** Unlike other frameworks, the role focuses one person on removing obstacles. This means that the team can concentrate on getting the job done.
2. **Dedicated coach:** The role gives one-person responsibility for coaching others. No one can “pass the buck” on this. Therefore, one person has the focus of helping all members of the organisation to understand the framework.
3. **Impartiality:** A scrum master can be as helpful to a team as a product owner (see below) without picking sides. The only focus is on making sure the framework and project is successful. This can help solve problems and gain trust.
4. **Responsibility for framework not delivery:** This is almost reverse psychology. The scrum master is only concerned with making sure the framework is carried out as the scrum rules say. Divorcing the responsibility for the framework from the responsibility to deliver means that he or she can concentrate on making sure that rules are followed which in turn creates a well-oiled machine. If the scrum master's job is done and everyone in the scrum team is performing their role, then the development team can deliver.
5. **No single point of control that could fail:** Since a scrum master does not control the team, the absence of one does not leave the team in disarray. The scrum master sets up a system that everyone can follow in his or her absence.

5 reason why THE PRODUCT OWNER ROLE works

Background

A product owner turns business objectives into requirements for a team in the form of a product backlog. Each item on the backlog is the description of a feature in the product. The product owner is responsible for maximising the return on investment for the business, managing the product backlog and making priority calls for the development team.

Reasons

1. Time maximised for business return on investment:
The product owner is not responsible for delivering the work or maintaining the process but simply for making priority calls and maintaining the requirements backlog. This allows a great deal of focus.
2. Dedicated source of requirements: There is no one else in the organisation that needs to be consulted on a project's requirements. Senior stakeholder requirements flow through the product owner for a single point of contact.
3. One person responsible for changes in requirements: As the business picture changes only one person needs to capture the new requirements and update them.
4. Achieves the best compromise: Even senior stakeholders will need to trust their product owner with the final decision. This aligns the business and makes appropriate compromises for the good of the product.
5. Aligns the customer and team, daily: This role is the interface between the business and the team. His or her presence at all the scrum meetings means that the team is always acting on the latest information.

5 reasons why THE DEVELOPMENT TEAM ROLE works

Background

Scrum development teams are cross functional (consisting of many skill sets working together) and self-organising (managing themselves to achieve the project goal). The team is made up of domain experts whose expertise is trusted to deliver the project. A scrum team should ideally be limited to between five and nine members to keep them lean, communicative and productive.

Reasons

1. A group of dedicated experts: Explicitly calling the team out as experts, means that scrum teams are assembled to solve problems on their own. This frees up other roles to focus on their own areas of expertise.
2. Flexible to business needs: Scrum teams adapt to a given situation in order to get a product increment built. Any decisions should be tied only to a business requirement. This in turn gives a business long and short-term flexibility and reduces wasted effort in favour of targeted effort.
3. Lean and cost effective: The small size combined with high degree of expertise means that things get done to a high degree of quality with minimal technical communication.
4. Less management needed: Teams organise themselves. This means that everyone else can concentrate on his or her own role.
5. Highly scalable when given the resource: Large teams can be separated and organised through regular meetings called scrum-of-scrums. The teams each have scrum masters to keep them coordinated. Caveat – when two or more teams work on the same code-base, the team will need to decide if this is feasible.

