HOW TO MAKE AGILE WORK IN OUTSOURCED PROJECTS?
CONTENTS

1. INTRODUCTION .......................... 3

2. WHAT IS AGILE? .......................... 3

   2.1. Benefits of agile ................. 4

   2.2. Make your agile outsourcing a success .... 6

3. WHAT IS SCRUM AND HOW IT WORKS? ....... 8

   3.1. The Scrum Theory ................. 9

   3.2. Roles in Scrum – Scrum Team .... 10

   3.3. How can Scrum Master be a support? .. 11

   3.4. Scrum events ....................... 12

   3.5. Are there alternatives to Scrum? .... 13

4. AGILE IN PRACTICE – WHAT ARE YOUR DOUBTS? .... 15

   4.1. What to do when agile doesn’t work? .... 15

   4.2. Agile in bigger teams ............... 16

   4.3. Agile learning curve .............. 17

5. CONCLUSION .......................... 18

WHO WE ARE ................................ 19
1. INTRODUCTION

Being a team doesn’t always come as easy as you’d expect. There are rules, timetables, and lots of pressure. However, imagine taking part in projects that are flexible, open to changes and are based on adaptive planning. This is all possible thanks to the implementation of agile.

In this whitepaper, we will show you all the merits of working in agile and introducing it to your outsourcing development process.

Additionally, the paper can be a guide for those who start to work with this methodology, since we’ll try to give you all the necessary information about its principles.

2. WHAT IS AGILE?

As Everette R. Keith from NYU writes:

Agile Software development philosophy has its roots in the reality of today’s markets. The emergence of agile software processes attempt to deal with the issues introduced by rapidly changing and unpredictable markets.

The approach is built on The Manifesto for Agile Software Development written in 2001. It lists rules of a proper agile methodology where constant improvement goes hand in hand with flexible changes, which both lead to adaptive planning.

What is more, working in agile helps to deliver products faster and react quicker to occurring changes.
2.1 BENEFITS OF AGILE

Do you often find yourself getting lost in all the details of your project? Does going through all of them drive you up the wall? Imagine you can have it all developed in iterations, much quicker than you could envision.

Impossible? Not anymore, once you experience the benefits of working in agile:

1. ACCELERATED TIME TO MARKET

Because of set time frames and goals, implementing agile usually shortens time to market. All because each released increment is a fully useable piece of software.

2. INCREASED PREDICTABILITY

Using agile’s practices and tools, it is possible to predict timelines and estimate the cost of the whole project. All this to determine whether or not you should continue the development.

3. BIGGER PRODUCTIVITY

Agile teams that maintain working code with every iteration tend to be more productive. Integration issues are discovered sooner, the overall product quality is higher, and the team works productively throughout the development cycle.

4. ENHANCED SOFTWARE QUALITY

Agile methodologies have a number of safeguards (e.g. taking advantage of automated testing) that help to reduce the number of problems. This allows to make adjustments if necessary and gives early sight of any quality issues.

5. EASE OF MANAGING CHANGING PRIORITIES

Due to short iterations you can quickly adjust the product to your needs and market opportunities, without disrupting the whole development process.

6. IMPROVED PROJECT VISIBILITY

Since goals are set for each iteration, you know exactly what to expect and when to expect it. You can also have insight into the project’s progress whenever you want.
Frequent interactions between the business and the development facilitate team’s work and assure that the scope of work is clear, whereas your involvement guarantees that the emphasis is absolutely on building the right product.

But don’t just take our word for it – VersionOne has released its 10th annual on State of Agile™ report. Between July and November 2015 they interviewed 3,880 individuals from a broad range of industries in the global software development community. They were asking about, among others, the reasons for adopting agile.

Isn’t getting your product faster, yet without compromising on quality exactly what you would envision while deciding on outsourcing? Once you apply agile methodologies to your software projects, you can have the projects done right.

Do you wonder how agile can improve your software outsourcing? Read on.

### REASONS FOR ADOPTING AGILE

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>62%</td>
<td>Accelerate Product Delivery</td>
</tr>
<tr>
<td>56%</td>
<td>Enhance Ability to Manage Changing Priorities</td>
</tr>
<tr>
<td>55%</td>
<td>Increase Productivity</td>
</tr>
<tr>
<td>47%</td>
<td>Enhance Software Quality</td>
</tr>
<tr>
<td>44%</td>
<td>Enhance Delivery Predictability</td>
</tr>
<tr>
<td>44%</td>
<td>Improve Business/IT Alignment</td>
</tr>
<tr>
<td>40%</td>
<td>Improve Project Visibility</td>
</tr>
</tbody>
</table>

*Source: VersionOne*
2.2 MAKE YOUR AGILE OUTSOURCING A SUCCESS

After years of the Waterfall approach, which is a sequential design process used in software development, it’s high time you thought about a change.

Why? Because the Waterfall approach consists of certain stages and only once those are completed, the developers move on to the next step. This means that the focus is solely on the tasks at hand, which may sometimes hinder cooperation. What is more, after a step has been completed, you can’t go back to a previous one - not without scratching the whole project and starting from the beginning.

Here comes another drawback - the possibility that the customer will be dissatisfied with the end product. Since the work is based upon documented requirements, a customer may not see what will be delivered until it’s almost finished.

It also takes a long time to produce all the necessary documentation at the very beginning which should be carefully followed and is necessary to plan the work.

Agile, on the other hand, is thought of as more team-based, supporting creativity and openness. What is more, it emphasizes rapid delivery of complete functional components. Rather than creating tasks and schedules, all time is ‘time-boxed’ into sprints.

But it is still not enough to ensure that once you implement agile, your project will be a success.

That is why we are here to give you five ideas on successful agile outsourcing.

### KNOW YOUR GOAL

Whenever you start a project you usually know what you want to achieve. It is the same with outsourcing. Ask yourself a few questions: Is there talent available locally? Is it possible to complete a project on a tight budget?

Any successful outsourcing project should start with a clearly outlined goal. Be precise about your expectations and share them with your outsourcing partner. To get the most of agile outsourcing, you must openly express your needs and objectives.

### COMMUNICATE

Communication matters. Outsourcing works well only when communication lines are open. Once you define your goals for the project, you should share your needs with your partner and receive meaningful feedback. The development will go smoothly and without problems if you keep in touch with your development team. However, if you find yourself out of time – designate a decision-making person, who will be available for the team throughout the project and will be responsible for communication between both sides.

Electronic means of communication are very helpful here, but don’t undervalue the power of face-to-face meetings.
TRUST YOUR PARTNER

It is hard to cooperate and achieve your set goals without trust. This is why you have to trust your software development partner. However, this trust must be earned. Look for the companies that are focused on long-term collaboration, but will let you build up the scale of your commitment or exit, if you’re not satisfied with the outcomes. It may also be a good idea to check their certifications e.g. (Certified Scrum Master or Certified Agile Professional).

Once these preconditions are met, do not hesitate to rely on the knowledge, experience, and expertise your outsourcing partner brings to the table. Established outsourcing companies will always aim to deliver on their promises and keep you satisfied with the outcomes, because this is the basis for successful long-term business relationships.

ENGAGE

Your outsourcing partner will take the workload off your hands, but this doesn’t mean the process will be effective without your engagement. It is your product that is to be developed, so you have to stay involved. The outsourcing company will surely bring expertise but, in the end, it is you who knows your business best and needs to make key decisions in order for the project to succeed.

BE OPEN. BE AGILE

As your business evolves, your demands are likely to change. Also, the solution you initially had in mind may not be the one you really need at the end. That is why you should stay as open-minded as possible about your project plans and embrace the change. Fine tuning the project’s scope and adapting it as you go are at the heart of the agile approach.

In the end these will be the key factors determining how easy it will be for both of you to work together. Which, in some way, makes it akin to marriage – so don’t forget to have some fun while working too.
3. **What is Scrum and how it works?**

Let us now tell you a little bit about Scrum, one of the most popular agile frameworks. It helps to address complex adaptive problems, while productively and creatively delivering quality products.

Scrum has been used to manage complex project development since the early 1990s. It is not a process or a technique for building products; rather, it is a framework within which you can employ various processes and techniques. Scrum clarifies the relative efficacy of your product management and development practices, so that you can improve them.
3.1 THE SCRUM THEORY

Scrum is founded on empirical process control theory, also called empiricism.

Empiricism asserts that knowledge comes from experience and that making decisions should be based on what is already known. Scrum employs an iterative, incremental approach to optimise predictability and control risk. This approach is based on three pillars, which uphold every implementation of empirical process control: transparency, inspection, and adaptation.

Let us briefly describe each of them:

- **TRANSPARENCY**
  One of the key advantages of using Scrum is that the progress of the work and the team is visible to all. It means that significant aspects of the process must be visible to people responsible for the outcome, which requires those aspects to be defined by a common standard and nomenclature.

In order for Scrum to be effective, each of the above-mentioned three pillars should be followed – one without the others may have a negative impact on the effectiveness of Scrum implementation.

- **INSPECTION**
  Scrum promotes frequent opportunities to review the Artifacts (Product Backlog, Sprint Backlog and Increment) in order to detect undesirable variances. But it also allows to monitor current state of the work and how the team is using Scrum. This gives insight into current challenges, progress made so far, and encourages the team to solve complex problems.

- **ADAPTATION**
  Once the results of Inspection show that one or more aspects of a process deviate outside acceptable limits, and that the resulting product will be unacceptable, it is crucial to make changes and adapt to new ideas. This adjustment must be done as soon as possible to minimise potential delays, enabling the team to reduce or even eliminate any issues that may hinder the development.
3.2 ROLES IN SCRUM — SCRUM TEAM

Scrum wouldn’t be possible without self-organising and cross-functional teams.

**Self-organising teams** choose the best way to do their work, rather than being directed by others.

**Cross-functional teams** have all the competencies necessary to accomplish the work without depending on others.

The team model in Scrum is designed to optimise flexibility, creativity and productivity.

**WHAT IS MORE, THERE ARE SPECIFIED ROLES WITHIN EACH TEAM:**

- **PRODUCT OWNER...**
  
  ...is responsible for maximising the value of the product and the work of the Development Team. They decide how the product will ultimately look like, as well as check if it is meaningful and well-matched to the requirements of users / customers. This is why it would be optimal if they were appointed by the client.

- **SCRUM MASTER...**
  
  ...is responsible for ensuring that Scrum is understood and followed. This is done by ensuring that Scrum Team adheres to the Scrum’s theory, practices, rules and values. Scrum Master is simply the manager and the guide of the process.

- **DEVELOPMENT TEAM...**
  
  ...consists of professionals who do the work of delivering a potentially releasable Increment of a product at the end of each Sprint. Development Teams are structured and empowered by their company to organise and manage their own work.
3.3 **HOW CAN SCRUM MASTER BE A SUPPORT?**

---

**THE PRODUCT OWNER?**

- Finding techniques for effective Product Backlog management
- Helping to clearly communicate the vision, objective and components of agility practices and Product Backlog items to the Development Team
- Teaching the whole Scrum team to construct clear and concise Product Backlog Items (PBI)

**THE ORGANISATION?**

- Leading the processes of Scrum adoption and coaching the people engaged in the process
- Helping employees and stakeholders understand and enact Scrum and empirical product development approach
- Working with other Scrum Masters to increase the effectiveness of the application of Scrum in the organisation

**THE DEVELOPMENT TEAM?**

- Coaching the Development Team in self-organisation, cross-functionality and creating high-value products
- Removing impediments to the Development Team’s progress
- Coaching the Development Team
3.4 Scrum Events

Scrum methodologies were created to make work easier and smoother. This is why all events are timeboxed. This creates regularity and minimises the need for meetings not defined in Scrum.

The heart of Scrum is a Sprint – as you’ve already read, it is a defined period of time, for which work is scheduled. During a Sprint, a ‘done’, useable, and potentially releasable product increment is created. A new Sprint starts immediately after the conclusion of the previous one. Once a Sprint begins, the duration of meetings is fixed and it cannot be shortened or lengthened. The remaining events may end whenever their purpose is achieved, ensuring an appropriate amount of time is spent without allowing waste in the process.

Sprint Goal, specifies an objective for the Sprint. It provides guidance to the Development Team and it is created during the planning meeting. It gives the team some flexibility regarding the functionality to be implemented. Items selected from Product Backlog deliver one coherent function, which can be the Sprint Goal.

Like the projects, Sprints are bound to accomplish something. This means that at the end of the project, as well as at the end of a particular Sprint, the development team usually provides a working product.

During the Sprint:

- No changes should be made that would endanger the Sprint Goal
- Quality goals do not decrease
- The scope may be clarified and re-negotiated between the Product Owner and Development Team as more is learned

**Sprint Planning...**

...allows the team to create a plan. It is timeboxed to a maximum of eight hours for a one-month Sprint and answers the following questions:

- What can be delivered in the Increment resulting from the upcoming Sprint?
- How will the work planned for the Increment be done?

**Sprint Review...**

...is the time when the team presents which Product Backlog items have been ‘done’ and discusses with the client what went well during the Sprint, what problems they ran into, and how those problems were solved.

It is vital for the client because they see what had been done during the Sprint and can adjust their plans for the future. It is the moment they can accept or suggest any changes to the ‘done’ Increment.

The client gets the review of the timeline, budget, potential capabilities and a time frame for the next anticipated release of the product.

**Sprint Retrospective...**

...is an opportunity for the Development Team to inspect itself and create a plan of improvements to be enacted during the next Sprint. Sprint Retrospective occurs after the Sprint Review and prior to the next Sprint Planning. This is a three-hour, time-boxed meeting for one-month Sprints.

**Daily Scrums...**

...are 15-minute, time-boxed events for the Development Team to synchronise activities and create a plan for the day.

During such a meeting each team member should answer the following questions:

- What did I do yesterday that helped the Development Team meet the Sprint Goal?
- What will I do today to help the Development Team meet the Sprint Goal?
- Do I see any impediment that prevents me, or the Development Team, from meeting the Sprint Goal?
3.5 ARE THERE ALTERNATIVES TO SCRUM?

In the world of IT you are surrounded by technical terminology and it may sometimes become too overwhelming, especially when you are looking for the best outsourcing company on the market and want your product developed with attention to detail.

Let us briefly present to you the agile alternatives to Scrum such as Kanban, Scrumban and Extreme Programming (XP).

3.5.1 KANBAN

Kanban is a technique for managing a software development process in a highly efficient way characterised mainly by lack of iterations. It underpins Toyota's 'just-in-time' production system, which is why it works best for support and maintenance projects, where work is planned only 2-4 weeks ahead, the response time is short (the team solves problems on the spot) and new tasks can be introduced at any time.

Kanban:
- No pre-defined roles
- Continuous flow
- Continuous delivery
- Cycle time

VISUALISE WORK

Creating a visual model of the workflow helps to make the process clear and observe how it moves through the system. Any bottlenecks are instantly spotted, which leads to increased communication and collaboration.

LIMIT WORK IN PROGRESS

It is the cornerstone of Kanban. Work-in-progress is limited and new elements are added only when there is available capacity within the limit. These constraints will quickly illuminate problem areas in the flow so they can be identified and resolved.

FOCUS ON FLOW

By observing how the work flows through the system, it is possible to analyse potential problem areas in which the flow is stalled and to implement changes. Then, the cycle is applied again to see what effect the changes had on the system, and so on.

CONTINUOUS IMPROVEMENT

Kanban allows the team to measure its effectiveness by tracking flow, quality or throughput. Further analysis can change the system to improve the team’s effectiveness.
3.5.2 Scrumban

Scrumban is a hybrid of Scrum and Kanban - it is actually a stage between a transition from Scrum to Kanban.

Some teams blend the ideals of Kanban and Scrum into ‘Scrumban’. They take fixed length sprints and roles from Scrum and the focus on work in progress limits and cycle time from Kanban.

Scrumban uses the planning on demand technique where iterations aren't strict or rigidly time-boxed which makes the development very efficient. Scrumban uses flow diagrams allowing to spot the vulnerabilities of the project at early stages.

But most importantly, it focuses on regular and frequent communication with the client who should be aware of the development and all the aspects of the project. That is why, it is necessary to appoint a contact person on the client’s side. This methodology may work for startups, fast-paced projects or support & maintenance ones.

Scrumban guarantees...
- ...continuous flow of work, where all team members are equally busy at all time because tasks are not assigned to a specific person
- ...monitoring of time which leads to balancing team’s capacity versus the demand
- ...continuous improvement where issues raised by the team can be brainstormed and resolved as soon as they appear
- ...saving time by using planning on demand technique, instead of having daily meetings or spending hours on task estimations.

3.5.3 Extreme Programming (XP)

The primary goal of XP is to develop high quality software more effectively and to be flexible to customer’s changing requirements. It is usually based on frequent releases in short development cycles, in order to increase productivity and introduce checkpoints at which new customer requirements can be adopted. In short, XP allows to deliver the software you need, as you need it.

The XP team...
- ...performs four basic activities within the software development process: coding, testing, listening, and designing
- ...works in very short development cycles which allow any potential changes to be incorporated quickly
- ...works on items in order of importance
- ...appreciates frequent and rapid feedback
- ...can introduce new items of work into an iteration and switch between them.

But enough theory.

You’ve learned a lot about agile itself, Scrum and its alternatives. Now let’s see how you can apply this knowledge in your workplace.

It is natural that during the decision stage you may have some doubts. Let us help you dispel them.
4. Agile in Practice - What Are Your Doubts?

If you are new to agile methodologies, it is understandable you have doubts and second-guess your decisions. However, your questions cannot stay unanswered.

4.1 What to Do When Agile Doesn’t Work?

Even though agile has become the word on everybody’s mouth in recent years and it is very easy to list all of its benefits, it doesn’t mean it will be the best choice in every situation. Why? Because certain things must be avoided to make it a smooth sailing.

So what should you focus on?

**Communication**

Communication is always the key to achieve your goals. In case of agile, communication between the client and the developers is the crucial factor, which cannot be overlooked. Outsourcing may present many challenges, clear understanding of the requirements being one of them. In case communication is not considered important, the development is bound to go astray.

That’s why it is crucial to express your needs clearly and explicitly, to make sure that both sides share a common vision of the project.

**Product Vision**

Speaking of the vision, development with the same overview in mind, is what you need to focus on. The goal of the product must be specified and known from the very beginning. Those outsourcing companies which do not pay attention to a client’s needs may put the product at risk and it will be much harder to work on.

That is why agile is based on sprints – these short periods of time usually help to check whether you are on the same side with your outsourcing partners, but without knowing where the vision’s heading, even sprints won’t bring anything to the table.

**Cross-Functional Teams**

The development team should be able to deliver a working software, which means they ought to have multiple skills.

So, rather than each team member having expert knowledge in one area, they all should be able to complement each other. Creating such a team may appear tough at first, however in time, most team members broaden their expertise and embrace the framework.
4.2 Agile in Bigger Teams

What about when the project is bigger and more complex? Should you avoid even trying agile in such a case? No, not at all. When multiple teams are working together on a project it is still possible to implement agile.

How? The answer is the Scrum of Scrums (SoS).

The Scrum of Scrums is a way of scaling Scrum, applied within bigger groups (over 12 people). It is based on dividing these teams into smaller (5-10 people), independent ones, with separate Scrum events which proceed just as described earlier. Each sub-team is to designate one person to communicate with the representatives from other teams. The Scrum of Scrums is simply one team composed of those representatives.

This way, all individual teams don’t cut into each other’s tasks and are aware of the overall progress of work, still being able to discuss common project’s matters. The primary purpose of the SoS is to support agile teams in collaborating and coordinating their work with others, increasing their efficiency.

…and this is what’s called the Scrum of Scrums and something that can help you make your agile work, regardless of team size.
As simple as it may sound, it is true. Adapting to the new system may be problematic at first and may even slow the team down. However after a few sprints everything will be easier and you will notice the benefits.

That is why applying a process that works for your organisation may take some adaptive learning.

As you can see all kinds of obstacles can be overcome and everything becomes easier in time, so don’t get discouraged at the beginning and give agile a chance. Because if you implement this methodology wisely, the investment will definitely pay off.

Are there any ways to shorten this learning curve? Of course there are:

### LEARNING BY TRIAL AND ERROR

As simple as it may sound, it is true. Adapting to the new system may be problematic at first and may even slow the team down. However after a few sprints everything will be easier and you will notice the benefits.

### TRAINING

Learning each new skill requires a course conducted by an experienced practitioner. It’s no different with agile. Take time to understand the rules. And remember that attending the training doesn’t necessarily mean you become an expert in a field right away. Introduce the ideas into your daily routine slowly and always remember that it takes a lot of work to master them.

### MUTUAL UNDERSTANDING

It is obvious that the development team must become experienced in agile in order to use it on a daily basis. Nonetheless, so should the client. It is no secret that mutual engagement is vital to successful work and release of a functional product. This is why the client is the very person who should understand the values of implementing agile, take part in the process and get accustomed to it.
5. CONCLUSION

Agile is becoming so popular due to its undeniable benefits, including its positive impact on product development.

Even though at first its implementation in software outsourcing may seem laborious, in time you will be able to complete your projects more smoothly.

All thanks to open communication mixed with commitment which give you more control over your development than any of the traditional project management techniques. This very approach unites the development team and the client, making them trust each other more and collaborate better. The work itself becomes more organised, as the expectations are known from the very beginning.

If you are after tangible results, transparency and partnership in your outsourced projects, agile should be your first choice.

Do you still have questions?

CONTACT US AND LET US ANSWER THEM
WHO WE ARE

Future Processing is a software company which, since its foundation in 2000, has grown from a group of a few friends into a team of over 900.

Our mission is to solve business problems of industry leaders worldwide with the use of the latest technology.

We help businesses by offering full cycle bespoke software and mobile application development, legacy software modernisation, database and security services, as well as analysis & design, apart from working in other areas.

We are ISO 27001:2013 certified and we have been a Microsoft Partner since 2007. In 2016, we were announced Outsourcing Service Provider of The Year 2016 by Global Sourcing Association (GSA).
ARE YOU CONSIDERING OUTSOURCING SOFTWARE DEVELOPMENT?

Feel free to contact us and let’s talk about how we can work together!

Future Processing
ul. Bojkowska 37A
44-100 Gliwice
POLAND

+48 32 461 23 00
sales@future-processing.com
www.future-processing.com