



From Waterfall to Agile development

Lessons learnt

Gustav Coleske – Scrum Master



Contents

Introduction	3
Identifying the Product owner and key users	4
Identifying the MVP vs. Blueprinting and functional/technical specifications	4
Requirements gathering	5
Security and Integration	5
Delivery	6



Introduction

We are increasingly delivering projects that are based on the Agile approach for our clients who have traditionally run projects that used the Waterfall approach. We delivered our projects by using Mendix which is a digital transformation company that has developed a platform as a service product for rapid application development. This cloud-based platform allows users to build, integrate and deploy web and mobile applications very quickly. These applications can be built either from the ground up or as a layer on top of existing legacy systems (see <https://en.wikipedia.org/wiki/Mendix>). We have noticed common themes throughout these projects that a Waterfall approach won't handle very well. We want to share some of the common things we have come across relating to the adoption of the Agile approach over the traditional Waterfall approach.

If you are new to application development, or do not understand the approaches that can be used, or find the whole thing a mystery you can find excellent information on the web. For a very quick and high level overview of which methodology to use see the diagrams below. From my experience it's the application that you are developing that should dictate the approach or methodology you should adopt and not try to fit a square peg in a round hole.

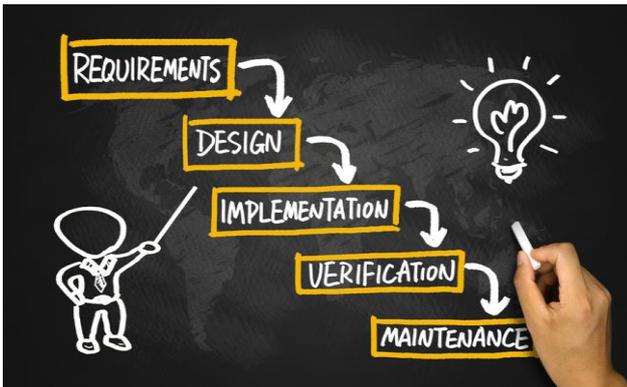


Figure 1: Waterfall



Figure 2 : Agile

If your project is very structured, such as a payroll system, it will be relatively static (changes are infrequent), with a clearly defined output against a clear set of rules. Or if there is little need for interaction between the developer and client team, then Waterfall is a good option.

Agile does not require the exact design to be completed before development starts. The premise is that you know what the application should do but work out how it should operate as you go. This is achieved through short and productive development with playback and then incorporating changes quickly – seeing is believing after all. One recent project I was lucky to work on included functionality that was not thought of at the beginning of the project, but we were able to incorporate it in the agreed project timelines.

Identifying the Product owner and key users



Identifying your product owner and typical users is critical. As with the waterfall approach getting the commitment from team members is critical. **However this is even more so with agile due to the short development and testing cycles.**

The business owner needs to commit time to attend daily scrum meetings as well as sprint planning and sprint demo sessions. The Product owner is the gatekeeper for the project and will be the contact for stakeholders, with decision-making power during the implementation, will respond to queries of the development team and will ensure the value of the work being done.

Whoever is identified as the product owner, will need to have capacity and accountability to assist with this.

Key users will assist in delivering the user stories and input for the development process. Once the user stories are developed, the key users will test and accept functionality based on these stories.

If there are issues with existing functionality or new ideas come to mind while testing, the key users will be raising the relevant feedback items.

As with any project, not getting the commitment from the relevant product owner and key users or changing these stakeholders midway through a project is recipe for failure.

Identifying the MVP vs. Blueprinting and functional/technical specifications

What is an MVP? To quote Eric Reiss in his article on “Startup Lessons learnt”- “The minimum viable product is that version of a new product which allows a team to collect the maximum amount of validated learning about clients with the least effort.” This is known as the MVP – Minimal Viable Product.

Identifying a product that is ideal for the Agile approach tends to be a bit of a problem for clients using the Waterfall approach. Clients using the Waterfall approach tend to want a product with all bells and whistles that cater for all their requirements at the end of the build and test phase (near the end of the project).

What projects are typical of the Waterfall approach and what projects are good for using the Agile approach?

Writing a complex payroll for a company is typically not a project undertaken using Agile as this cannot be implemented in stages. There are simply too many complex rules that cannot be implemented and the product should be a final version catering for all these rules. Writing an application that allows a new joiner of a company to upload eligibility to work documents, which are uploaded to the company’s HCM system, on the other hand is an ideal candidate for this. It’s a function that can be implemented quickly and can evolve to include additional functionality for new joiners, which can be implemented in subsequent phases.

Sometimes project stakeholders can lose focus of the outcomes of the MVP. Once the MVP is identified, it is important to focus the product team on what should be achieved to resolve this business problem.

Requirements gathering

With the Waterfall approach, requirements are fixed and very expensive to change the further the project progresses. This works well if clients know exactly what they want, developers know exactly how to build it and things don't change along the way.

Unfortunately this is true in very few cases. The reality is that clients discover what they want, developers discover how to build it and many things change along the way. Agile caters for this by allowing changes to be reprioritised in every sprint. Changes to user stories in the backlog are allowed at any time. During the sprint planning session clients decide what they want to include in the next iteration (whether this is existing or new stories in the backlog) and what to include in a development sprint. Priorities of user stories may change and the Agile approach allows for this.

For the right type of project, comparing the time taken to capture user stories and sprint planning with the typical time it takes to run workshops, blueprinting and writing of functional and technical specifications for waterfall, the Agile approach is markedly faster.



Security and integration



The speed of delivery when using a RAD tool like Mendix is often grossly underestimated. This proved to be particularly true when building applications that integrate with other systems. If a product integrates with an ERP or HCM system for example, it is essential that the integration points are setup, connections of the system through firewalls and middleware connections and web services are **setup and running early in the project** Due to the speed of development there is a potential danger of being held up while they are being set up. In a waterfall approach which may take longer overall, security and integration can be addressed later, which is not the case with agile.

Getting input from key people can be challenging especially taking into account their busy day jobs. Buy-in from the relevant parties around timelines and focus on the speed of delivery in these timelines **before** the project commences is critical.

Delivery

The reaction during implementation is typically positive and users from the traditional approach of using the waterfall approach are usually very positive.

Users tend to see a working product within the first few days, which in most cases is not something they have come across before.

Typical comments are generally:

- “Do you have a number of developers working on this in the background?” or
- “Is this really a working system or is it smoke and mirrors?” or
- “Have you developed this before the project started?”.



We are all passionate about providing our customers with a good experience and using Agile and Rapid Application Development platforms is quite good at giving clients a good experience and getting positive feedback.

Contact:

David Helme
EPI-USE Ltd
Brook House
10 Church Terrace
Richmond
Surrey
TW10 6SE

David.helme@epiuse.com
Tel: 020 8332 8787
www.memdixsolutions.com

Mendix Solutions is EPI-USE UK's agile development business unit.

Thank you for taking the time to read this paper.

If you would like discuss the projects undertaken by **Saga Healthcare** or **Clarks International** or if you have an idea that you would like to explore please get in touch. You maybe also interested in our **Embrace Digital Applications to Drive Business Performance - But How?** paper.

