



W

W

O

S

Rob Cole

Your quick start
introduction to *agile*

!

g

e

Praise for *Agile Now*

‘A practical look at what it means to be agile and how this isn’t just for the techies among us. Straightforward guidance on the theory, tools and, most importantly, the themes of mindset and culture were handled with humour and honesty.’

**Amanda Hickey, Head of Business Change
and Innovation, Solera-Autodata**

‘An easy read for anyone curious about how agile can be applied to any business environment. It’s more than enough to get anyone up to speed and it doesn’t over complicate the subject.’

David Daiches, COO and Co-Founder, INSHUR

‘Agile is an extremely lucrative market, and this book unlocks its secrets for just the cost of a takeaway pizza. This humorous, easy-to-read book upskills its readers and entertains them as it educates them on the project management method, which is all the rage in large and small companies up and down the land. It brings its readers from being on the outside to very much on the inside for this much sought-after skill.’

Gerry McLaughlin, owner, www.ITContractor.com

‘Agile is perhaps the most misunderstood, misused and confusing term in project management today but here, finally, is a book that talks plainly and simply about agile - I especially love the ‘Go Agile in 15 minute’ suggestions. Read it! Go agile!’

Peter Taylor, author, *The Lazy Project Manager*

‘As a doctor working in the NHS for 38 years, agile would benefit doctors and managers engaged with managing the world’s largest bureaucracy. It provides focus on clarity of thought with distinct aims and objectives rather than process and form which can stifle innovation and positive change.’

**Professor Neil S Tolley MD FRCS DLO, Head of
Otolaryngology – Head & Neck Surgery -Imperial
College NHS Healthcare Trust, London, UK**

Although there's never a right answer when it comes to putting things in an order of importance, it's prudent to be able to walk before trying to run. So, a good variety of accommodation plus a check-in facility is always going to be higher up the wish list than a communal hot tub. There's no need to fret about getting the priorities spot on up front because the running order can be easily changed, with items promoted or demoted when necessary.

The AgileParks *Backlog* is a living thing and it's a healthy sign if stories are added and priorities change. Early on, the management team may realise it's impossible to operate without an office and a small temporary portacabin becomes a new requirement with a higher priority. Or customer surveys may indicate cosy lodges will be much more popular than anticipated and that gets moved up the pecking order too.

Sizing is important

An important think-piece when assessing relative priorities is the comparative size of each piece of work. When a job that delivers high value is up against one with little return then the decision is an easy one, but life is rarely that simple. Usually it's a choice between something with higher net worth that will cost more versus a quicker win that's less useful but cheaper to deliver.

This can be a minefield and tangible costs are usually the main things considered – basically any hard cash involved and number of hours of graft required. Predicting the fixed costs and anticipating the number of person hours to deliver certainly helps in deciding what to do first. However, there's a tendency to get bogged down in the calculations. This approach is time-consuming and the costs are only indicative anyway.

Agile estimation uses a different approach. Rather than chase the impossible goal of predicting exact costs, the aim is to get an indication of the relative size of work within the *Backlog*. Is *User Story X* bigger than *Y*? Is *Y* larger or smaller than *Z* and the size of each is described in a non-financial way. A very popular way of doing this is by assigning T-shirt sizes: XS, S, M, L, XL, XXL and even XXXL if you wish!

Crucially, with *agile*, estimates are always produced by the people who'll be doing the work. Elements considered do not include the

potential effort required but also the complexity of the story and the number of unknown factors surrounding it.

Agile in action

AgileParcs size estimates

Is the tent pitch job bigger than the one for building the wooden lodge? Where does the admin block/convenience store fit into the scale? And what about those new-fangled yurts? The important thing is to define the relative size of planned work in the *Backlog*, as follows:

- Tent pitches – S
- Waste bins – XS
- Shower/toilet block – XXL
- Recycling facilities – XS
- Yurts – M
- Entrance signage – XS
- Reception/admin building/shop – XL
- Wooden lodges – XL
- Mobile homes – L
- Caravans – L
- Entertainment centre/bar – XXXL
- Communal hot tub – M
- Swimming pool – XXL

Don't be surprised if you don't agree with these initial estimates. The sizing exercise always sparks valuable debate.

When there are differences of opinion it usually points to misunderstandings about exactly what is involved or highlights unknown factors. Is it best for AgileParcs to deliver the caravans, mobile homes and wooden lodges one at a time or in batches of two or three? Are the

lodges going to be handcrafted from scratch or shipped in as a package and then assembled? If the admin block is bigger, can it incorporate a convenience store, general storage and even a coffee shop or bar?

The generalness of the T-shirt assessments avoids prolonged debate about minor points of detail. Once *User Stories* are identified as S, M, L or whatever, then this assessment is a useful factor in juggling priorities. Business value is the primary driver but there are occasions when an XXXL valuation may tip the balance in favour of other low-hanging fruit.

Go agile in 15 minutes

Build a prioritised *Backlog*

After grabbing a coffee, quickly brainstorm a list of the things you need to do today. Don't agonise, get everything down – even the nice-to-haves-if-time-permits. Check that every task produces something explicit that will be of use in its own right.

Put your to-do list into running order and let gut feel be your guide. Then tag each item with a T-shirt size of S, M, L or XL and have another think in case there are any quick wins which deserve pushing up the pecking order, or complex tasks that are candidates for demotion.

Think about what you absolutely must do before going home and check it's an achievable target. If it is, then you've pinned down the minimum you need to deliver before heading off with a clear conscience, and with plenty more to do if time permits.

Voilà, a prioritised Backlog.

Deliver in bite-size chunks

Before *agile* came along, the most popular approach to project planning was to spend ages detailing a huge long list of requirements and then insist they're all essential from day one. This included endless deliberation about all the bells, whistles and other

nice-to-haves which could take months or even years in extreme cases. Many great ideas died a slow and painful death this way even before any real work started.

Of course, everything isn't really needed up front. But in the bad old days, anyone with any nous knew that the first delivery was probably all you'll get apart from fixes to any glaring snags. There was little point in hoping for anything that wasn't very clearly specified in the contract signed up front. To make matters worse, there was a long, long, long wait for anything to turn up and hell to pay if any changes were asked for along the way.

In stark contrast, *agile* delivers smaller chunks regularly. Starting with the most important stuff first and then working through the wish list, with plenty of opportunity to change track when necessary. A constant stream of smaller deliveries with plenty of scope for a rethink whenever needed. This is referred to as *Incremental Delivery*, with each bundle building on the previous one.

Agile definitions

Definition of Done

The AgileParcs *Vision* sets up the big picture but it isn't enough to judge when each individual task is complete. As a general rule of thumb, for any job to be considered done at the end of the workflow, the end product must be fit-for-purpose and ready to go. There were different opinions about what that meant in practice.

The AgileParcs team drafted a *Definition of Done* which applies to all their work. It included a series of conditions to keep the design team, construction, sales and marketing, internal maintenance and finance teams happy. Of course, the main focus was on the hard graft, but all the touches needed to make it a saleable commodity were there too.

All sorts of boring wrap-up activities and admin were included as well. The site must be cleared, tools and equipment tidied away, timesheets filled in and handed over to the finance

department. Plus, they added on a lighter note, a celebratory cup of tea and a bit of cake for the workers in the staff canteen.

The Definition of Done spells out without ambiguity how to confirm each task is complete. There's no set format and it's always a collaborative effort.

Build a foundation

The first no-frills package that goes out is the foundation stone of the *Vision* and is known as the *Minimum Viable Product*, usually abbreviated to *MVP*. The jargon is a bit mind-boggling but the underlying concept is simple enough: it's the bare essentials needed to get going, with the intention to kick on from there. Using this approach, it's cheaper to get started and easier to change direction if things don't pan out quite as expected.

The tricky question is what's the minimum needed to get started. Defining the *MVP* isn't always plain sailing and is often a hot potato. Of course, the smaller the *MVP*, the earlier the first usable chunk arrives. But it must be fully functional too and it must be viable as a standalone product or service. It's certainly fine for AgileParcs to start with only tent pitches and mobile homes but there would be no point in launching with an admin block but no accommodation.

Agile definitions

The *Minimum Viable Product*

The majority of *agile* terminology slips off the tongue and is close to self-explanatory. One big exception to the rule is *Minimum Viable Product (MVP)* which is both a mouthful and often misrepresented. In an attempt to avoid overcomplicating this concept, let's say the common usage is *just enough core features to be useful*.

Developing a complex product or a service takes time and is costly. Far better to launch with the bare minimum needed and see if it hits the mark. This way, the time to market is considerably reduced and it's easy to see if things are on the right track. If they are, additional features can be added over time and if not, it's easy to change course.

Getting the balance right between too much and too little is one of the biggest challenges. It's important to be ruthless in identifying the *MVP* without going overboard. Pulling any item out should have severe consequences and effectively prevent the ship from setting sail. Anything with an acceptable temporary workaround shouldn't be included initially.

There's likely to be a sharp intake of breath once this is pinned down. There's scope for manoeuvre but always ask whether it's best to get the must-haves out quickly or to wait longer for a more wide-ranging offering.

With a fully functioning physical or digital task board in place, it's much easier to stand back and reflect on the minimum requirements. It's easy to shuffle around the *User Stories* and discuss permutations with others. When doing this, always bear in mind that once the *MVP* is launched, other stuff will be coming soon afterwards. Normally it's better to keep the *MVP* slimline and accept some short-term pain to get off to a fast start.

Beware the *agile* workflow tripwires!

There are risks with any new endeavour and when things go badly wrong it usually starts to unravel right at the start. It's important to set up a solid foundation to avoid going off piste early on. Be mindful of the following:

- **Lack of Vision:** It's easy to underestimate the importance of defining what, who and why. It's time well spent and if it's not a straightforward exercise then the alarm bells are ringing.