

[More](#) [Next Blog»](#)[allankellynet@gmail.com](mailto:allankellynet@gmail.com) [New Post](#) [Design](#) [Sign Out](#)

# allan's blog - Agile & Digital Business

I help companies and teams that create software.

Saturday, May 13, 2017

## Why do devs hate Agile?



Someone asked me this the other day: "Why do devs hate agile?" and as I worked through my answer I thought it worth writing down. There are three reasons I can think of but before we get to them...

First off, I don't think all "devs do hate Agile". Or rather, I don't think the vast majority of them hate it - and hate is a very strong word. Sure some do, no doubt but all? A blanket "all devs hate" ? No.

I do think its is fashionable to slag agile off, and few do this better than the agile community themselves. And unfortunately like a lot of fashions once it gets started it catches on. Once dev A says "I hate Agile" then dev B thinks its cool to say it too, and dev C sees A and B doing it so joins in.

The ironic thing about all this is that Agile is the product of developers. In the beginning it was developers - like me - who saw that the classical way of doing things (write down what the customer wants, design it, plan it, code, it, etc.) didn't work very well but noticed that the way most work actually happens (quick, code foo... O foo is not quite right change it, quick) was actually better.

I was inspired by Jim McCarthy's book Dynamics of Software Development. Jim was a developer, albeit a developer who had started managing other developers. It was that book that made me think "alternative ways are valid."

Similar insights were occurring with developers all over the world - as they raced to fix Y2K using heavy weight processes. Some of these coders went on to become mildly famous: Kent Beck, Ward Cunningham and Martin Fowler for example.

And here in lies number #1 why "devs hate agile."

Theft and imposition.

Agile was all about dev in the early days - especially XP. Just 10 years ago coders would say "I'd love to work Agile by my (project) manager won't let me." So we set about making Agile manager friendly, we expanded the business arguments for why Agile was good, and managers got it. Now you hear developers saying things like "My manager wants me to work agile but he doesn't understand" or "My manager wants me to work agile but doesn't help."

In the beginning Agile was a bottom up movement. Now it is a top-down movement, change is imposed on people rather than people wanting to change.

That is wrong.

Making it worse is the fact that many of those imposing the change frequently fail to understand the change they are imposing or do not question their own thinking. Management thinking needs to change too - start with Software Development is Upside Down.

### Reason #1: Agile is now an imposed change.

In my experience developers want to work Agile because true Agile allows them - no,

#### Contributors

- [Allan Kelly](#)
- [allankelly2015](#)

#### My books

- [Continuous Digital: the alternative to projects](#)
- [Xanpan](#)
- [Little Book of User Stories](#)
- [Business Patterns](#)
- [Changing Software Development](#)
- [Agile Reader](#)

#### Newsletter sign-up

[Sign-up for my newsletter](#)

#### I am at...

##### Upcoming events with Allan Kelly

**Planning for value**  
Private: 15 September at Oxford

**Agile Cambridge**  
Software Acumen: 27-29  
September at Cambridge

**Agile Bristol**  
Agile Bristol: 12 September  
Evening at TBA

**Agile Dice Game**  
TechCityCoffee: 25 July 2017  
2pm-4.30pm at Near Old Street  
Roundabout

[widge@surfing-waves.com](mailto:widge@surfing-waves.com)

#### Speaking events

- [Upcoming events](#)
- [Events RSS feed](#)



#### More

- [OnTwitter: @allankelly.net](#)
- [My Homepages: allankelly.net](#)

#### Twitter feed

**demands** - they do a quality job. Agile doesn't deliver half the benefits it promises if organizations don't pay attention to quality, that means their technical practices, that's what we used to call technical excellence. That means developers doing work that are proud of.

Namely "technical practices from XP", specifically: simple design, relentless refactoring, test driven development (plus behaviour driven development), pair programming and things like face-to-face conversations and "story is a placeholder for a conversation."

I'll point the finger specifically at Scrum, and later Kanban, for not mandating these practices - something I did with Xanpan. Part of making agile acceptable to management was removing the words extreme and programming, and down playing the difference high quality can make.

Without these practices teams are driven harder to deliver something sooner and quality drops. As quality drops it gets more difficult to deliver anything in a short amount of time. Consequently more is demanded of coders, stress and tension rise and it is not fun.

**Reason #2: Agile without technical quality makes developers lives worse.**

I remember meeting some coders in Cambridge, almost the first thing they told me was "We hate Agile, our managers went on a Scrum course and have insisted we do it for months." I quickly discovered that they were not doing any technical practices, as a result they were racking up more and more technical liabilities and making their own lives harder. Once I explained that they were missing the technical practices their attitude changed.

Now to the final reason, perhaps the big reason...

The Agile toolset is intended to help teams organize themselves, it is intended to make problems visible so they can be addressed and fixed. In the hands of people with the right attitude this is brilliant.

Teams don't need a manager (although one may still be useful).

Teams can see problems.

And teams can fix problems.

But... the very same tools used by someone with the wrong attitude are a micro-managers dream.

*Which micro-managers wouldn't want everyone to give a status report at 9.00am every day?*

*Who wouldn't want to see all work broken down to pieces for which NAMED individuals could be held accountable?*

*And why wouldn't they want to make a shocked face and send a very clear "that is not acceptable" message every time an estimate was high?*

Visibility becomes a tool of blame.

I once helped a team at an airline set up a Kanban board, instead of using it to see bottlenecks, problems and find opportunities to improve the managers concerned used it to assign blame, point fingers and demonstrate that nothing was happening because someone else wasn't doing their job.

**Reason #3: In the wrong hands the same Agile tools are very effective micromangement tools.**

Posted by Allan Kelly at 8:40 am



Tweets by @allankellynet

allan kelly Retweeted



**Anna Filina**  
@afilina

Pro tip: if your method is 225 lines long, then you should probably refactor your code ASAP. I frown at methods over 50 lines.

18h

allan kelly Retweeted



**Multinewmedia**  
@Multinewmedia

Get the new book from Allan Kelly (@allankellynet) for just \$5 on LeanPub for

Embed

View on Twitter

Follow by Email

Submit

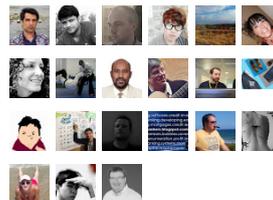
Subscribe To

Posts

Comments

Followers

Followers (39) [Next](#)



Follow

Alltop



Popular Posts



Software has diseconomies of scale - not economies of scale

Some things can never be spoken

Programmers without TDD will be unemployable by 2022 (a prediction)



Why do devs hate Agile?

Dear boy, have you ever tried programming?

Banking systems stink - the pain of an botched HSBC release

Blog Archive

2017 (27)

July (3)

June (4)

May (3)

Why do devs hate Agile?

Clinic: Product Owner/Manager question

Heavy lifting is the easy bit

7 comments



Add a comment as Allan Kelly

Top comments



**Emilie Esposito** 1 day ago · Shared publicly

Thank you for this very good article! I strongly agree. The #3 struck me like thunder, because I've never seen it phrased so clearly. Adopting an agile method without adopting the agile mindset can be dangerous. Micro-management is at the opposite from agile mindset.

[Read more](#)

· Reply



**Dov TSAL SELA** 2 months ago · Shared publicly

Now repeat after me:  
"Developers hate Agile because what they call Agile is not Agile."

+2 · Reply



**Allan Kelly** 2 months ago

Interesting observation Dov, how would you describe the thing that is not Agile but which developers call Agile? - I think you've got something here and what to follow your thinking



**Emilie Esposito** 1 day ago

+**Allan Kelly** Maybe Cargo cult agile :) When companies or teams imitate practices they don't understand, rather than embracing agile mindset and then build the practices suiting their needs.



**Csaba Sarkadi** 2 months ago · Shared publicly

The biggest thought here: "My manager wants me to work agile but he doesn't understand" or "My manager wants me to work agile but doesn't help."

I would like to add also, that what I have experienced, most problem is that Agile  
[Read more](#)

· Reply



**Allan Kelly** 3 weeks ago

Csaba, I agree and disagree with you. I agree that if an organization is to achieve the full promise of Agile then other parts of the company need to change too, and I agree that many of them don't.  
[Read more](#)



**Nicolas U** 2 months ago · Shared publicly

Excellent blog post, about the manager friendly agile.

[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)

- ▶ [April](#) (5)
- ▶ [March](#) (4)
- ▶ [February](#) (3)
- ▶ [January](#) (5)
- ▶ [2016](#) (43)
- ▶ [2015](#) (42)
- ▶ [2014](#) (41)
- ▶ [2013](#) (43)
- ▶ [2012](#) (45)
- ▶ [2011](#) (53)
- ▶ [2010](#) (70)
- ▶ [2009](#) (90)
- ▶ [2008](#) (85)
- ▶ [2007](#) (79)
- ▶ [2006](#) (77)
- ▶ [2005](#) (62)

JavaCodeGeeks



Syndicated to Java Code Geeks