are projects with high developer turn over rate really a bad thing? [closed]

Asked 14 years, 7 months ago Modified 14 years, 5 months ago

Viewed 620 times

▲

13

▼

Closed. This question is opinion-based. It is not currently accepting answers.

Want to improve this question? Update the question so it can be answered with facts and citations by editing this post.

Closed 5 years ago.

Improve this question

I've inherited a lot of web projects that experienced high developer turn over rates. Sometimes these web projects are a horrible patchwork of band aid solutions. Other times they can be somewhat maintainable mosaics of half-done features each built with a different architectural style. Everytime I inherit these projects, I wish the previous developers could explain to me why things got so bad.

What puzzles me is the reaction of the owners (either a manager, a middle man company, or a client). They seem to think, "Well, if you leave, I'll find another developer, because you're expendable." Or they think, "Oh, it costs that much money to refactor the system? I know another developer who can do it at half the price. I'll hire him if I can't afford you." I'm guessing that the high developer turn over rate is related to the owner's mentality of "My ideas are always great ideas, and if you don't agree, I'll find another (possibly cheaper) developer who agrees with me and does what I want". For the owners, the approach seems to work because their business is thriving. Unfortunately, it's no fun for developers because they go AWOL after 3-4 months of working with poor code, strict timelines, and insufficient client feedback.

So my question is the following:

Are the following symptoms of a project really such a bad thing for business?

- high developer turn over rate
- poorly built technology - often a patchwork of different and inappropriately used architectural styles
- owners without a clear roadmap for their web project, and they request features on a whim

I've seen numerous businesses prosper with the symptoms above. **So as a programmer, even though my instincts tell me the above points are terrible, I need to take a step back and ask, "are things really that bad in the grand scheme of things?"** If not, I will re-evaluate my approach to these projects..ie. Do I build long term solutions or band-aid solutions?

** At the risk of this post being closed as non-programming related, I'd like to argue that I think it is programming related because answers to this question will influence the way a developer approaches a project. He will have a better feel for how far in advance he should plan his development (ie. build short term or long term solution) knowing he may quit at any moment.

refactoring project-management entropy

Share

Improve this question

Follow

edited Jul 19, 2010 at 8:18

asked Apr 30, 2010 at 0:51

learningtech

33.6k ● 86 ● 265 ● 441

9 Answers

Sorted by: Highest score (default)

▲

8

▼

✓

All three symptoms are bad. They really are a bad thing for business. That being said:

Software development exists to make tools. That's it. It's not an end, in and of itself - you're a tool maker.

There are very successful businesses that operate using poor tools. They may not be run as well as they should be, but good results can, and often do, come from bad tools. Also remember, though, that eliminating your three symptoms will likely make the company even more effective, especially in the long term.

Share Improve this answer

Follow

answered Apr 30, 2010 at 0:56

Reed Copsey

564k ● 80 ● 1.2k ● 1.4k

If the only *uses* software (and is not in the business of *making* software), then it only has to be good enough to support whatever it is the company really does.
– Philip Kelley Jul 15, 2010 at 16:53

6

High dev turnover is a symptom, not a cause. The cause is bad management. If those businesses prosper, it's usually in the short term and usually precedes a buyout, a merger, or an outright failure. I've seen it happen over and over.

Share Improve this answer

Follow

edited Apr 30, 2010 at 2:48

answered Apr 30, 2010 at 0:57

Robusto

31.9k ● 8 ● 58 ● 78

cannot up vote this enough, unfortunately I would assume not many managers read stackoverflow – mxmissile Jul 15, 2010 at 16:34

2

If you can afford - run. There are bad companies out there but there are good ones too - at least better than the mess you describe.

Share Improve this answer

Follow

answered Apr 30, 2010 at 0:56

Otávio Décio

74.2k ● 18 ● 165 ● 228

2

All those 3 things are not good let me focus on turnover. I'm seeing it happen right now. management/company are being cheap so they don't care much about the team, techonology or process, just the bottomline. So in turn (eventually) team members don't care about the project, just THEIR bottomline. After several months, they decide it's not worth the stress and move on. We are a small team of 6 developers, this year 3 people want out and it's just July. 2 people came in, one more is coming. Seems all we're doing is transition and project turnover. Team does not mature and is ineffective. our customer senses this, and instead of giving the team more projects (more money for company) they limit it to certains apps. I wonder when management will realize that being cheap is costly!

Share Improve this answer

Follow

answered Jul 15, 2010 at 16:13

javacruiser

65 ● 3 ● 10

2

If I may take a Devil's Advocate view on this for a moment:

- Some people like a challenge. Achieving extremely difficult things are very exciting for some people and there are some developers that enjoy finding those uber hard problems and work on those. Having something difficult to do appeals to some people.
- The turnover means that each time someone is starting from scratch rather than retaining all the ideas and thoughts that the previous developer had in building out the software, whatever it was intended to do. Sometimes multiple heads can make a good thing. After all, how many people developed Windows 7? ;)
- The poorly built point is where someone may think, "Oh, I can shine here by fixing some of this stuff," and at times it can work for a while. Ka-ching!
- The lack of a roadmap and almost advocating the "Cowboy coding" style may appeal to those that want great autonomy and just move to their own beat. After all, who needs methodologies and best practices when one has supernatural powers to use to make this awesome stuff that will take no time at all?
- There is the question of what is the root cause of the turn over rate for developers? Is it just that the project is killing developers or the pay is so bad almost anywhere else would be better or something

> else? Just something to ponder here as there can be many a way to get rid of a developer, both literally and figuratively.

To be serious about this for a moment, there are some people that do enjoy high pressure situations and others that want to avoid them at all costs. Most people are somwhere between the two extremes. Where do you think you fall on that scale though?

Share Improve this answer Follow

answered Jul 15, 2010 at 16:28

JB King

11.9k ● 4 ● 40 ● 49

2 I like high pressure situations only if my clients appreciate my efforts. "That's amazing! No one was ever able to fix that problem, let alone do it in 1 night" is a much better reception than "Pff...that's what I pay you to do...so did you check out that new photocopier we got?". – learningtech Jul 16, 2010 at 15:27

2

I'll address each of your three points in turn. High turnover in any industry is considered bad for business and a management problem. However, I've read several books about corporate politics and cultures and the effect those have on the corporate bottom line. One book I read studied several major corporations over a 20-year span. It found that poisonous cultures grow slowy and tend to be "lagging indicators" of bottom line performance problems. It also found that when some of the companies were able to hire new CEO's who ultimately "turned the ship around", it took 10 - 15 YEARS to stop the bleeding. So in a VERY big picture view, yes turnover is poisonous, although it truly is a symptom of the larger problem. A symptom that should not be ignored. (Even though it usually is ignored for long periods of time. Ever notice that it takes HR a very long time to realize that a department's turnover might be tied to a bad manager?)

Poorly built technical infrastructure - or products that are sold to customers are obviously bad for the bottom line. I think that only non-technical people fail to understand this. Of course there is a range between "not optimal but works" and "barely works as long as you restore the database once a week it gets us by". I think the reason this happens is that the cost portion of the "holy trinity" is always chosen in favor of quality. In my experience this is guaranteed to be a hard and fast rule. If management has to choose between cost, quality and schedule, quality is always the first tossed to the wayside.

The problem of owners without a clear roadmap and feature creep are a symptom of lack of business discipline. Feature creep costs money. And when it's bad enough, it can actually prevent anything from being completed.

Share Improve this answer Follow

answered Jul 15, 2010 at 17:14

Carrie Cobol

90 ● 6

1

The interesting thing to me about your question is that you say that they are thriving as a company, so it makes me wonder if the technology is as important to them. Maybe the problem is that they don't see the value in better technology (and they might be right in their case, I'm not sure what kind of business they are).

Share Improve this answer Follow

answered Apr 30, 2010 at 1:02

david
33.5k ● 12 ● 67 ● 88

I've worked for companies in the real-estate industry that had such high margins that they could screw around endlessly and still make a ton of money. – Greg Jul 15, 2010 at 16:51

1

In general, a very high turn over rate for employees isn't good in any company. When it comes to software, high developer turn over is bad because of all the tutoring that has to be done for the new one, and the "big picture" knowledge that goes out of the door. So if software is important for the business, high turnover rate is bad for business.

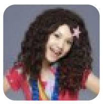
Only doing requested features without a roadmap is a one way path to bloatware. If you have no clear strategy, goal or purpose for a product, your only source for what to do is customer requests, which might be bad. This is so because the customers might actually not know what they want, thereby requesting features they won't use.

Share Improve this answer

Follow

edited Apr 30, 2010 at 1:02

answered Apr 30, 2010 at 0:57



Lars Andren

8,751 ● 7 ● 42 ● 57

1

From one perspective, the attitude(s) you're quoting are understandable. Software development isn't cheap, and most people/businesses are trying to save money everywhere they can. However I think they're usually shooting themselves in the foot with this sort of behavior.

One suggestion for dealing with this is to get a copy of The Mythical Man Month and read the section on why adding more programmers to a late project will only make it later (it's the title - and second (in my copy) - essay). Many of the same ideas apply to replacing a developer ... except that if you're working solo, it's likely you may as well start over, as figuring out what the previous person did may take longer than starting from scratch. After you've read the essay, give a copy to anyone who's taking the attitude you cite and ask them to read it. No guarantee that it will help, but it's worth a try.

Share Improve this answer

Follow

edited Jul 19, 2010 at 3:49

answered Jul 15, 2010 at 16:45

GreenMatt

18.6k ● 9 ● 55 ● 81