PDSA.com

Solutions for the Real World

PDSA Special Report

Estimating Software Projects

Estimating Is Hard, but it Doesn't Have to Be

Estimating software development projects is one of the hardest things to do. First, programmers just don't like doing it. And why should they? They are usually incorrect and may feel the heat when their estimates slip. Secondly, most programmers just don't have a method or process by which to develop an estimate. In this special report I will lay out some simple steps to follow to help you provide estimates. If you come up with a process for estimating, you will be on your way to being right a large percentage of the time.

If you ask three programmers to give you an estimate for a project, you will have three different answers. Why? Each one uses their own background, experience, and checklist to derive an estimate. So what do you do? Leverage those experiences in a structured approach and document it. What we did at my company PDSA, Inc. (www.pdsa.com) is develop an estimating model and approach to estimating. We built and defined an estimating process. Remember: you can never improve an estimating process if it is in only in someone's mind. But if you build an estimating process and document it, you can improve the process and measure it. Once you have done that, you can begin to build solid and more predictable estimates.

Step 1: Build an estimating model.

What are all the elements to an estimate? Collect your teammates together and brainstorm all the elements of a project. There are many books out in the market, and I am sure your teammates will have some ideas to. Here are some sample elements: logical database design, physical database design, unit testing, integrated testing, prototyping, requirements meetings, coding, etc. You will likely have a list of over a hundred items. Classify them into categories like: requirements, development, testing, customer buyoff, etc.

Step 2: Use the estimating model

Does that sound silly? Why did I just go through all the work in Step 1 only to not use it in Step 2? That is because your customers or upper management will want a quick answer. The problem is that they will remember your quick "answer." Try and resist. Once you start using your estimating model and you can show how successful you are, they will trust and believe in you. Now that is something new for IT!

Step 3: Track your work

An estimate is only just an estimate unless you have time tracking in place. I don't mean the kind where I track whether you went to lunch or not, but the kind of tracking that captures for every single task in your estimating model how many actual hours are spent. This is important. You have no idea how good your estimate really is unless you track actuals against it. You must do this and I can't impress this point upon you more. By understanding your actuals you can refine your estimating model. Refining your model may include changing the number of hours you think a task really takes, or it could be that you forgot several tasks. Now you can add those new tasks back into your model. It is imperative to leverage your lessons learned and improve your model. Better models enable you to better predict project results, which yields in projects coming in on time and your customer's appreciating it.

Once you have estimated and tracked several projects, you will begin building up a repository of estimating information. Another benefit is that at times a custom may ask for a quick estimate. You don't have time to do much if any analysis, but they need that magic "number" to put into their budget. You could mine that data to find projects roughly similar to the new project. You could then at least provide a range of estimates based on some very high level project criteria. The estimate certainly is not perfect and should come with the usual disclaimers, but at least you have used real data based on your experience. At PDSA, we have over 15 years of tracked data and it has been extremely valuable in helping our clients in the initial planning and budgeting phase.

Estimating "Rules of thumb"

Estimates must be created using a standard, repeatable process. But when developing your estimating model, your model should produce reasonable estimates. If your model produces estimates that are unreachable, your development team will not work within your process. So your estimates need to be reasonable (within the context of your company, skill sets, leadership, etc.). On the other hand, make sure your estimates are a bit of a stretch. Each person should be challenged and your software development process should be challenged as well. As you challenge your team and processes you will find a wonderful outcome: innovation! You will innovate and discover new or different ways to do things better, faster, and cheaper. It may sound like a cliché, but it works.

Summary

Estimating is an important skill that can be learned and improved. It is not a scientific process unless you let it become one. Accurate and consistent estimating builds customer loyalty and team confidence. Tracking provides real time feedback to not only your team members to see exactly where they are, but also to your customers.

If you want more detailed information on estimating and other IT management practices check out the PDSA Agile ALM product (http://www.pdsa.com/agile). This product has all of the ideas expressed in this special report, and more, in one easy-to-use package.

Contact Information

If you would like to know more about the information in this special report, please contact either Paul D. Sheriff or Michael Krasowski at PDSA.

Paul Sheriff

(615) 675-4632

PSheriff@pdsa.com

Michael Krasowski

(714) 734-9792 x223

Michaelk@pdsa.com

Company Information

PDSA, Inc.
17852 17th Street
Suite 205
Tustin, CA 92780

Tel (714) 734-9792
Fax (714) 734-9793
www.pdsa.com

PDSA.com
Solutions for the Real World