

Complete Projects On Time

The Five Principles to meet your budget and deadlines

Beth Schechner

Elementool

The content of this eBook is provided to you for free by Elementool. You may distribute this eBook to anyone you know. If you quote or use the text in this eBook, we ask that you give us credit

www.elementool.com

Complete Projects On Time

The Five Principles to meet your budget & deadlines

- 92% of large scale IT projects fail.ⁱ

According to the market research company, Research and Markets, US spending on IT products and services is forecasted to grow to \$514.5 billion in 2010ⁱⁱ. However, the estimated cost of IT failures to the US economy is \$1 trillion!ⁱⁱⁱ Now we're talking about IT system failures and their net affect on the companies that rely on those systems. But you can bet money on the fact that the institutions and companies experiencing these losses know the IT company who built their system. Furthermore, they're not going to be buying from that company again, much less recommending it to their partners and contacts. In any company, large or small, your reputation is your biggest asset – you want to be in that 8% group. If your IT projects are failing, then you are not going to win the market share you need to compete in that \$514.5 billion market!

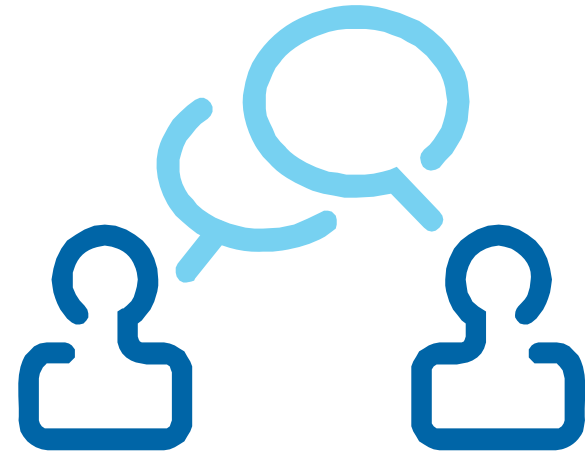
The following 5 principles lay out the steps you should be following, to make sure that your projects are completed on time, and on budget, successfully. Read them, learn them, know them, teach them and then make them stick.

Principle One: Communicate

- *40% of project managers cited poor communication as the leading cause if IT project failures^{iv}*

If everyone involved in your IT project doesn't know what they're working on, when it's due, how to get it done and who the audience is, how can you possibly expect your project to succeed? All of the other

principles that come later on rely on your communication, expectations, goals, resources, deadlines, priorities, reports and budgets all need to be available to your team so that they can do their jobs properly. Yes, communicating all of these things can sometimes be difficult, particularly if it's a short project with a tight deadline, or half of your team is located overseas. It doesn't matter. With all of the technology readily available to us today, there is NO excuse for lack of communication. Tools like web-conferencing actually make it possible for your developer in Hyderabad to share his screen with you in New York, to demonstrate how exactly to recreate major bug in your application.



If you make it easy for your team to communicate with management, and with each other, they WILL be more productive. If you put up unnecessary roadblocks in communication, you are preventing your team from working cooperatively and making it more difficult for your product to be released on time. However, using every tool at your disposal to encourage communication and knowledge sharing will absolutely save time when it comes to your project lifecycle.

If your developers and testers are able to easily communicate and view results from their team members, they can quickly incorporate feedback and issue fixes. As a manager, you should want to do everything you can, and use every tool at your disposal, to facilitate communication between and within teams. Ongoing feedback on the status of bugs and issues and individual test case steps during the Software Development Lifecycle (SDLC), can and will significantly reduce the amount of time your application spends in development.

Principle Two: Consolidate

- *Among all IT development projects, 84 % exceed acceptable constraints of cost, time and quality^x*
- *The average person gets 1 interruption every 8 minutes, or 50-60 per day.*
- *The average interruption takes 5 minutes, totaling about 50% of the average workday.*
- *80% of those interruptions are typically rated as "little value" or "no value" creating approximately 3 hours of wasted time per day.^{vi}*

We're talking about information and tasks here. Think about your typical day and how often you stop in the middle of one task to do something else; be honest. How much time do you spend looking for one specific email, out of the hundreds in your in-box? Searching for files on your hard drive after forgetting which folder you saved the one you need in? Opening files to find the most recent version or worse yet, working off of an old file version? Asking around for clarification on an issue? Writing reports? Almost every company can improve their process, cut down on tasks and put data within easy reach, but it isn't easy and will require change.

Make a policy of saving all company data in a common location, be it a shared server or a cloud application. Enforce this rule. Your project is much more likely to be completed on time if all the relevant project data is easily available and everyone knows where to find it.

Just take a second and think about how much time your team will save every day if they don't have to search for files or through emails. Reports could be pulled together quickly, background information on issues or test cases would be right in there with the initial issue report, and your project probably wouldn't run over time or budget.

These are small activities that usually take a few minutes but when you sum them up, at the end of the month, you'll find out that a lot of time is wasted. For example: on average your team members spend 15 minutes a day searching for files. This adds up to 5 hours a month of wasted time for each person in your team on this task alone.

Principle Three: Prioritize

You'd be surprised at how much time you, and your development team, are wasting by not assigning priority and severity to every task. Issues come in, bugs are discovered, and new application features are conceived of every day. Every item won't be crucial to fulfilling your business goals, but your team has no way of concretely knowing which tasks are most important to the company unless someone in management tells them. Priority tells your team how important it is that they resolve this issue immediately. Assigning priority and enforcing it, whenever any task or issue is assigned, will make your team work more efficiently.

Setting priority is akin to designating a time slot for every task that you assign a team member during the day. Urgent issues and tasks are certain to be tackled first thing in the morning, with low priority items waiting until later on when all others have been completed. With a clear idea of which tasks, issues or tests are most important, your team members can easily plan their days to make the best use of their time. Think about your own work habits. Without prioritizing, it's easy to get caught up in busy work, fielding every new issue that comes in. But just because an issue was the last one to cross your desk

Priority:

Immediate – High importance, this issue should be resolved immediately!

High – Should be resolved as soon as possible.

Medium – Should be resolved as soon as any more serious items have been addressed.

Low – This can wait until all higher priority items have been fixed. or until your next release.

doesn't make it the most important. Now think about how you structure your own to-do list? If you think that something is critical, but potentially time consuming, are you more or less likely to put it first on your list? However, if you KNOW that this thing is of highest importance to your boss, does it move up on your to-do list?

Severity:

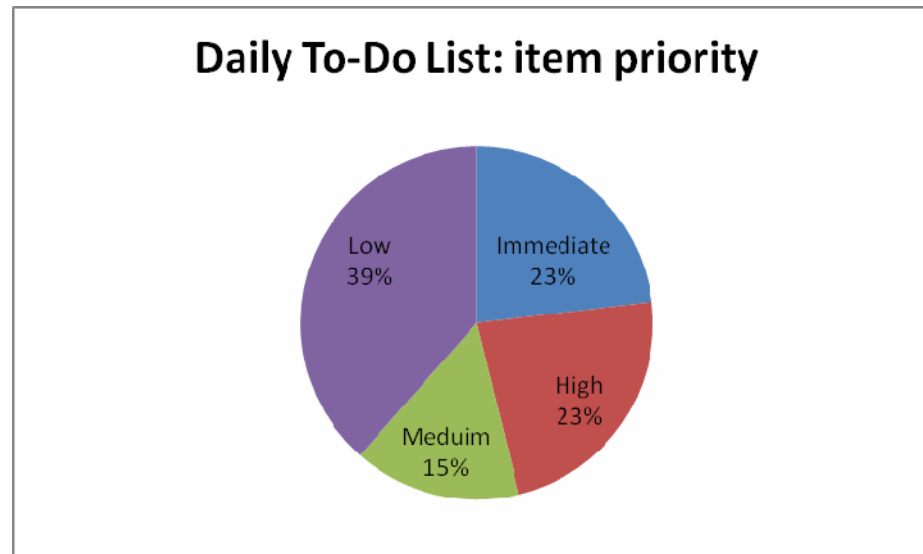
Showstopper– Is causing a complete failure of the system or program.

Major – Doesn't cause failure but does impair usability.

Minor – Loss of function with an easy work around.

Severity tells your team how serious the issue or bug is, based on how much it affects your software system. The severity of a bug gives it a more detailed ranking on your team to-do list. If you have 3 high priority issues, then how do you know where to start? By designating severity as well as priority, you're giving your team further direction so they can spend their time working on the most important tasks and get the best results.

Teams that know what is expected of them can quickly and efficiently go about their work. It's that simple.



Principle Four: Set Requirements

- *Companies pay 60% more on time and budget using poor requirements practices on projects.*
- *Over 41% of the IT budget at an average company is wasted by poor requirements.^{vii}*

Requirements Management is the first step of [Application Lifecycle Management](#) (ALM). Here, it's the fourth principle. In ALM and your SDLC, before you can do anything else, you need to identify, prioritize and agree on application (and project) requirements. Your requirements are the items that your application or project needs to include in order to meet your business goals. Once these basic steps are handled, they need to be communicated (!!)

back to all relevant parties including both development and business management. Requirements management is continuous throughout a project; and poor requirements management can lead to all sorts of delays and added costs the entire way.

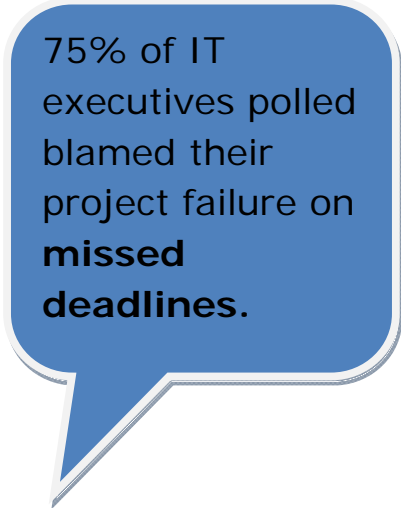
Obviously, according to the definition above, if your finished product doesn't meet your requirements, then your project has failed. So good requirements are one point here, but so are consistent requirements. During testing, you're checking the application against those requirements, to verify whether or not each individual feature is working properly. If business management is constantly changing tactics and requirements, your development team is going to be wasting their time testing and re-testing with every change. When requirements change frequently throughout a project, it's not uncommon for there to be a miscommunication over what features are wanted and expected. Overlooked or ignored requirements can also result in software application bugs - time schedule limitations (the next principle) will occasionally cause development teams to leave out previously defined requirements.

If requirements are unclear and imprecise, then team members may misinterpret them, and complete their tasks with a different vision in mind than what you had originally conceived of. As a result, you'll end up trashing what they just spent all that time on, and start all over again. Your team is now spending twice as much time on one task, because of unclear requirements, and doubling your project development time.

Principle Five: Manage Time

- *62% of organizations experienced IT projects that failed to meet their schedules.^{viii}*
- *84% of all IT development projects fail due to constraints of cost, time and quality.^{ix}*

Time management is the last, and trickiest to successfully implement, of the 5 principles. Effective time management takes a dedicated effort from business management and from every individual involved in your project. This step is the culmination of all of the previous principles: requirements management, prioritization, communication,



75% of IT executives polled blamed their project failure on **missed deadlines.**

and consolidation of information and tasks, into an effective work plan. As management, it's up to you to take the first step in setting reasonable deadlines and milestones for your team throughout the project. The key word here is reasonable. There are many available tools for tracking how long it takes to do certain tasks. If you utilize these tools on every development project that your organization takes on, after a short period of time you'll have an educated estimate of how long these development tasks typically take your developers to complete. Set these estimated milestones into your project timeline and CHECK IN! If you are not up to date on where your team is at each major milestone, than you cannot adjust your resources and schedules for bottlenecks and major bugs. Be assured, these bottlenecks and major bugs are GOING to happen, but if you're prepared and know early on, then they won't derail your entire project.

Assignment Summary Actual:

Website		Mar 22	23	24	25	26	27	28	29	30	31	Apr 01	02	03	04	05
[-]	Customer Account															
[-]	Inventory List															
[-]	Orders															
[-]	Shopping Cart															

Assignment Summary Estimate:

Website		Mar 22	23	24	25	26	27	28	29	30	31	Apr 01	02	03	04	05
[-]	Customer Account															
[-]	Inventory List															
[-]	Orders															
[-]	Shopping Cart															

Now that you've set project deadlines and individual milestones, you NEED to enforce them. Yes, delays happen, but if every developer and tester on your team knows that each task they're assigned has a set

deadline attached to it, they're going to work get it done on time. As each milestone is completed and checked off your master list, you get an emerging picture of your larger project and how it's meeting business goals.

Go Forth, and Conquer

Following and adhering to these 5 principles will put you in the elite 8%. In any company, large or small, your reputation is your biggest asset – you want to be in that 8% group. You can't fake it though; set the standards, find the tools to make them work and then enforce them throughout your team. Simply giving lip service to these principles won't put your project in the top group with those rare successful projects. Make them a part of your daily processes, get your team and other managers involved, and watch how well you meet your deadlines.

With 10 years of experience working with the world's largest companies, we've seen savings of up to 40% on project costs, by employing these 5 principles. To learn more on how to make them work for your company, go to www.elementool.com.

ⁱ Accenture: http://www.accenture.com/Global/Research_and_Insights/Outlook/By_Alphabet/InformationFail.htm

ⁱⁱ Research and Markets: United States Information Technology Report Q12010 http://www.researchandmarkets.com/research/98b3f0/united_states_info

ⁱⁱⁱ Simple Architecture for Complex Enterprises: <http://simplearchitectures.blogspot.com/2009/11/it-complexity-crisis-danger-and.html>

^{iv} The BULL Survey, 1998: [http://www.it-cortex.com/Stat_Failure_Cause.htm#The%20Bull%20Survey%20\(1998\)](http://www.it-cortex.com/Stat_Failure_Cause.htm#The%20Bull%20Survey%20(1998))

^v Accenture: http://www.accenture.com/Global/Research_and_Insights/Outlook/By_Alphabet/InformationFail.htm

^{vi} <http://www.shirleyfinelee.com/MgmtStats.htm>

^{vii} IAG Consulting: *The Impact of Business Requirements on the Success of Technology Projects*

<http://www.iag.biz/images/resources/iag%20business%20analysis%20benchmark%20-%20full%20report.pdf>

^{viii} Dynamic Markets Limited: http://advice.cio.com/remi/two_reasons_why_it_projects_continue_to_fail

^{ix} Accenture: http://www.accenture.com/Global/Research_and_Insights/Outlook/By_Alphabet/InformationFail.htm