Controlling Project Scope

How to resist the temptation to "boil the ocean."

By Dave Nielsen, PMP

The most common reason for software project failure is not failure to control budget or schedule, but failure to control requirements, either at the outset or during the build phase. Software projects are unlike those that deliver a tangible product such as a building, a bridge, or a dam. These projects receive much more attention during the planning phase and tend to be controlled with much more rigour than are software projects. You'll never hear of a plan to build a 20,000 square foot shoe store ballooning into a project to build a 10 acre shopping mall! Yet that's exactly what happens to a lot of software projects. They start off with one objective in mind but after the last stakeholder has added his or her wish list they bear no resemblance to the original vision. I call this dispersion of focused effort "ocean boiling." It's not an original term, but I think it fits this situation perfectly. Attempts to boil the ocean will expend a great deal of heat without changing the sea state. Attempts to deliver an extensive wish list of software features will spend a great deal of cash without changing the business.

Failure to maintain a narrow focus on the original objective of the project leads to overruns of both schedule and budget. I'm using the original budget and schedule here as we frequently see projects which have their budget and schedule changed during the course of the project, using the appropriate change management processes labelled as cost overruns and late deliveries based on their deviation from the original plan. Clearly, whether these projects are "on budget" and "on schedule" is a debatable point. The real point here is that a certain degree of deviation from the original budget and schedule is acceptable, but when the scope of a project experiences a significant increase budget and schedule are sure to increase beyond what all stakeholders will find acceptable. Think how many times you've read or heard about overruns on government projects in the media. The media rarely mentions whether these "overruns" are the result of approved change requests; their only point of reference is the originally announced schedule and budget and anything more is an overrun. You may not be working on a government project, but it is still important to curtail project scope for the organisation you are working for.

This article is not meant to be a comprehensive course on scope management, merely to offer a few helpful tips that will help you avoid managing a project that is perceived by your stakeholders to overrun budget and schedule. Those of you who haven't invested in project management certification should take a good PMP course or other PMP exam preparation training product and get certified. Doing so will give you all the tools and techniques you'll need to manage your project's scope. In the meantime, here a few tips you can start using right away.

Don't take the entire burden of curtailing project scope upon yourself. You share responsibility for this with your project's executive sponsor and you won't be able to control scope growth without their help. You need their help with the Change Control Board (CCB) when senior stakeholders come to you with additions to project scope, which they believe have merit. It is up to you to verify their business case and provide an accurate cost estimate and it is up to the executive sponsor and CCB to reject the change. You'll help with this decision if you can manage to provide an accurate estimate of the additional time and money that the change would cost. A finish date beyond what the organisation finds acceptable or a budget increase beyond the organisation's ability to pay will help the sponsor and CCB make the right decision.

Be as specific as possible in the Project Charter, preliminary Scope Statement, or Statement of Work (SOW) as possible. The Project Charter and preliminary Scope Statement will constitute the blueprints for the final product. Research usage of the software well before finalising these documents so that all the potential users, customers, or partners of the tool are consulted and their requirements captured. These documents should then become the benchmarks of your project's scope. New requirements that don't support the original intent should be discouraged. Features or functions which would better support the original concept should be given serious consideration. Ones that add significantly to the
original scope should not. I regard a request to support a new category of users, or market, to the scope as examples of requests for changes that are not appropriate.

Additions to the software system which are unacceptable to your project may be ideal candidates for the next project. Software systems are never static; they are constantly changing and evolving so they should have a process for capturing new features or fixes for the next system release. Ensure that your project has a direct connection between its change management system and the production change management system. Changes that would add too much to the scope of the current project, but that are otherwise supported by a sound business case should be assigned to the production feature pool. This approach has two advantages: it doesn’t lose sight of the value-add feature being requested and it isn’t an outright rejection of the requested change.

Don’t forget that the cost of the project must include project management activities such as project communications, meetings, workshops and other management functions. Your organisation may already have a well defined management methodology which you are expected to follow, otherwise you should be managing it using the best practices in the industry (the Project Management Body of Knowledge, or PMBOK 4th Edition describes these very well). Make certain that the work that you plan to do to manage the project is captured in the Project Charter and/or preliminary Scope Statement. Requests for additional work should be treated the same way as other requests to expand the scope.

A Statement of Work (SOW) will usually accompany an RFP or RFQ when work is being done by a vendor for a client. The SOW will usually be much more precise and detailed in its description of the work to be done, but not always. Make sure that the SOW you’re working from is detailed and includes a description of the project management functions and artifacts you will be responsible for. Duties, meetings, reports, and communications that were not identified in the SOW should be addressed through change requests. Change requests which would significantly alter the scope of the project should not be an issue. If it does become an issue, it will be the customer’s issue providing you have a well prepared SOW.

Verify that you can deliver the scope of the project as it has been defined for the budget allocated to the project and in the time given. This will probably be relatively easy when the project is similar to ones routinely undertaken by your organisation, or for smaller projects. More thought should be given to the project approach when tackling a very large complex project for the first time. Pilot projects and prototypes can be used to learn more about the cost and effort necessary to deliver the project. They will also prove the estimation techniques you have used, or suggest corrections to them when they prove to be inadequate. The work of these projects should never be wasted. The pilot should deliver a subset of functions for the planned system and the prototype should be a base that the project can be built on. Build the pilot or prototype into the overall project plan as the first phase. Review the feasibility of the project after this phase is over and re-calibrate your estimating methods if necessary.

Your first line of defence against budget and schedule overruns on a software project is a well drafted Project Charter and/or preliminary Scope Statement. These documents don’t have to contain a great deal of detail, that’s not their purpose, but do have to contain a description of the system’s functionality at a high level. Don’t try to make the documents detailed, but do make them precise. Next, make sure you’ve got all the right stakeholders and capture clear, concise requirements. Use a pilot project to prove your approach or a build a prototype to prove the technology. In both cases you should re-calibrate your estimation technique if necessary and ensure that you have the right schedule and budget for the project. Your last line of defence is your executive sponsor. The sponsor should be making the right calls on the proposed changes. Proposed changes that would increase the budget beyond the organisation’s ability to pay or extend the schedule beyond the time the organisation can afford to wait should be rejected. You can help by providing an accurate estimate of the cost of the change and the potential impact on the schedule. One last piece of advice: whenever the scope of your project is changed, make very certain that the new budget and schedule baselines are well communicated, then use them as your new baseline.

The tips and tricks described in this article are intended to help the project manager in using the best practices promoted by the PMI. Project managers who are certified have already implemented those best practices. If you haven’t been certified as a PMP (Project Management Professional) by the PMI and would like to learn more about certification, visit the three O Project Solutions website at: http://www.threoo.ca. Three O Project Solutions also offers a downloadable software based training tool that has prepared project managers around the world to pass their PMP exams. For more information about this product, AceIt, visit the three O website at: http://www.threoo.ca /featuress8.php.

© Project Smart 2000-2010. All rights reserved.