Avoiding Failure with Higher Education Technology Projects

I am frequently asked for a definition of a “successful” technology project. As a career senior technology executive, university educator, and now university chief executive, I have a deceptively simple answer. A successful technology project is one that is delivered on time, that comes within budget, and that meets or exceeds stakeholders’ expectations. Yet according to a study conducted by McKinsey & Company in collaboration with the University of Oxford: “On average, large IT projects run 45 percent over budget and 7 percent over time, while delivering 56 percent less value than predicted.” When I look around higher education, I would say these numbers are optimistic.

Why Higher Ed Technology Projects Fail

The easy answer to explain why technology projects in higher education fail is to place blame on ineffective project management and lack of communication. Technology project postmortems generally fail to get to the root causes of project failure—probably because true reflection means having to deal with the painful realization that the institution was ill-equipped to undertake the project in the first place. From nearly four decades of technology project-management experience, I see five main risk factors that lead to technology project failure. These risk factors are interrelated, and a failed project typically exhibits two or more of these factors.

Inadequate or Incomplete Definition of Requirements

In this age of agile project management, we seem to have lost appreciation for having a requirements document that details such items as the purpose for the technology project (including financial ROI), mandatory and desired functionality, and data conversion and retention requirements. In essence, what are the institutional, functional, and/or programmatic outcomes that the technology project must achieve? These outcomes form the basis for a project rubric, which can be used to evaluate aspects such as competing technologies (or systems), mode of implementation (e.g., “build versus buy” or a local server-based solution versus a cloud-based one), conversion schemes, documentation, and training. Without this rubric, how does one know whether or not this technology project has a chance of succeeding?

From nearly four decades of technology project-management experience, I see five main risk factors that lead to technology project failure.

Lack of Stakeholder Involvement

I cannot overemphasize the importance of stakeholder involvement in a technology project. All too often, the technology department of a college or university initiates a technology project—and obtains funding for it—without involving administration, faculty, staff, students, and others who will potentially be affected by the outcomes of the technology project. Collaboration and cooperation between stakeholders and the technology organization are keys to project success.

Two decades ago, I was engaged by a college to “rescue” a student information system (SIS) conversion that was late and over budget. It was in month eight of what was supposed to be a nine-month project, yet no academic or cocurricular departments knew anything about the project. They were not involved in the selection of the new system, were never scheduled for training, were never asked to validate the student data being converted, and were never included in any other aspect of the project. The technology organization’s rationale for this lack of stakeholder involvement went something like this: “They are too busy to be involved. We will train them when the technology team is ready to deliver the new SIS.”

In another, more recent SIS implementation, the institution’s technology organization proceeded with a “dry conversion” from a legacy homegrown system to an integrated vendor-supplied system. Thirty months later, and eighteen months after “completing” the SIS implementation, the institution is still struggling with the new system. Why? Without stakeholder involvement up front and during the project, the new SIS was made to mimic inefficient workflows based on the legacy system, data interrelationships were not understood by the technology folks (resulting in numerous data-related issues), and stakeholders again received “just in time” training that was ineffective.

Unrealistic Schedule

Higher education is not alone in its tendency to set schedules at the top of the organization. Some schedules reflect the reasonable constraints of a semester or term system; for example, upgrading computer lab equipment over spring break, implementing a new financial system based on the fiscal year, or deploying a new admissions system over a semester break. Fitting implementation into the first available break in the
academic or operating schedule is not a valid reason to rush a technology project.

Many higher education administrators (like their counterparts in the private sector) are unfamiliar with what it takes to deliver a technology project, especially the time needed to perform data quality control and train faculty and staff to a level of proficiency with the new technology. Yes, taking longer to correctly complete a technology project has an associated cost, but so does delivering one that is doomed to fail. As I used to tell my software engineering students: Spending $1 to catch and correct an issue in the requirements stage of a project will avoid the $1,000 that will be required if the issue is left undetected until after implementation.

Scope Creep and Inadequate Change Control
Without a project rubric, it is difficult to contain the scope of a technology project. With overactive stakeholder involvement, there is a tendency to add functions and features—or to turn on options—that at best are a marginal improvement to the system being delivered. The results are cost overruns, missed project deliverables, and schedule changes. Every technology project should have a formal change-control process to handle implementation realities and stakeholder requests. One reasonable way to deal with requested changes is to create a priority list of those requests that can be accommodated in the initial implementation and those that will come later.

Ineffective Documentation and Training
The project rubric should be the foundation for ensuring the adequacy and effectiveness of documentation and training. Vendor documentation and training should be examined for every function and feature listed in the project rubric; institution-developed documentation and training should emanate from the project rubric. It's never too early to start scheduling training for stakeholders based on their need to know or use the new technology. Here again, collaboration is essential.

Honing a Successful Technology Project Team
Mitigating project risk factors is a major part of avoiding technology project failures, but doing so will not be enough. A successful project requires strong project-management skills, frequent and clear communications with stakeholders, and a well-functioning project team. Honing a successful team to undertake a technology project requires preparation, leadership, and internal communication.

A technology project team brings together people who may or may not have worked together before. Some come from the technology organization, some are stakeholders, and still others are consultants or vendor representatives. It is extremely important that every member of the team knows his or her role and responsibilities and how to communicate within the team and has received an overview of the project itself, including goals, assumptions, limitations, constraints, deliverables, and deadlines. Conveying this information is the job of the project manager. Regardless of how many times these team members have worked together, this orientation is absolutely necessary.

Also key to preparing the project team is providing team members with the resources they will need to undertake the project—for example, hardware, software, Internet access, documentation, and training. Too often, higher education technology projects launch with insufficient resources, in part due to budgetary constraints. Time is another needed resource. Team members must have the dedicated release time necessary to spend on the project. This is extremely important for faculty and staff stakeholders, who will find it difficult to juggle project duties with everyday teaching or office responsibilities.

When a problem arises with the project—and it will—the team members and the project manager need to know about it and work together to get the project back on course. The project manager must anticipate problems, take corrective action, and help the team learn from the problems and issues encountered. Protecting the team from untoward external influence or pressure is also a key role for the project manager.

Continuous, positive reinforcement for team members can go a long way to moving the project forward successfully. There can be a lot of excitement and enjoyment in achieving the smallest of outcomes on a technology project. Acknowledgment of hitting project milestones helps build team morale, especially when the final deliverable is not yet in sight.

Takeaway
So what is the best way to avoid technology project failures in higher education?

- Have a strong project rubric based on stakeholder involvement. It will be the foundation for the project plan, documentation, and training, as well as ongoing communication with the stakeholders.
- Create a realistic schedule for the project and equip the project team with the necessary resources for success, including dedicated release time for this project.
- Commit stakeholder resources for testing and training.
- Empower the project manager to move the project forward without untoward pressure to change project scope or deliverables.

Finally, communicate ... communicate ... communicate!

Note

Norbert J. Kubilus (nkubilus@coleman.edu) is president and CEO of Coleman University, a private nonprofit teaching university. Founded in 1963 and located in San Diego, California, Coleman University offers degree programs that prepare its graduates for technology-focused careers.

© 2016 Norbert J. Kubilus. The text of this article is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.