CLASSROOM 2.0 THE BOOK
web 2.0 | social media | & more

5th Anniversary Book Project

12 Potential Education Technology Implementation Pitfalls and How to Avoid Them

By: Kelly Walsh

Creative Commons License: CC BY-NC-SA

Author contact:

Author Biography: Kelly Walsh is Chief Information Officer at The College of Westchester, a Bachelor Degree granting private institution in White Plains, New York. Walsh is also founder and author of EmergingEdTech.com, a blog and website dedicated to engaging students and improving learning outcomes with the use of Internet and instructional technologies. In addition to publishing hundreds of education technology related articles on the popular website, Walsh has also conducted a variety of presentations and workshops at industry conferences and institutions of higher education and online, covering topics such as "Flipping the Classroom: Succeeding with Reverse Instruction", "Using Popular Social Media Tools in Instruction", and "Fun, Free Digital Presentation Tools for the Classroom".

Activity Summary

There are many potential challenges that those responsible for implementing new or updated education and instructional technologies may encounter. This article discusses a dozen of these common pitfalls and offers suggestions for how to avoid or remediate these potential problems. Whether you are involved in a technology rollout on a small scale, perhaps in a single class or course, or on a larger scale such as across an entire institution, the observations in this article can help you to be equipped for success.

Class or subject area: Education Technology Implementation

Grade level(s): All

Specific learning objectives:
- Understanding commonly encountered education technology implementation issues
- Learning about various ways to avoid or deal with those potential problems

After many years of managing technology projects, I have experienced a good number of successes, but have also had the good fortune of being involved in some failures. Indeed, failure is an opportunity – a chance to learn how one might avoid repeating those oversights that led to failure in the first place.

Experience, coupled with formal learning about the project management process, has made me aware of a number of consistent potential pitfalls that often accompany many technology implementation projects. Following are twelve possible problem points for most education technology rollout efforts, and some thoughts on how to avoid falling into these traps (or at least how to be better prepared for them). These are really just fundamental concepts from the discipline of project management, adapted to instructional technology projects and the educational environment.

- *Insufficient Budget*
- *Failure to secure senior administrator or executive level buy-in*
- *Not having a sponsor*
- *Inadequate training*
- *Insufficient technical testing and validation*
- *Failing to involve faculty from the onset*
- *Forgetting to consider the student's perspective*
- *Relevancy (will the intended audience want to use it?)*
- *Not planning for adequate support*
- *Failure to incorporate training into on-boarding for new staff/faculty/students*
- *No risk assessment*
- *Limited (or missing) long term planning*

Some of these issues can affect your project from the onset, some can cause strife during the roll out phase, and some can turn a successful technology introduction into a less successful one over the long term.

This is not intended to be the definitive listing of education or instructional technology project failure points – it is simply a list of things to consider and plan for when working to implement new or enhanced technologies in the academic institution. Some of these concepts can easily overlap with others in the listing. These ideas can relate just as well to hardware or software projects, communications tools or instructional applications, and so on.

Let's explore each of these issues more closely and consider ways to avoid allowing them to negatively impact your technology roll out.

No 1: Insufficient Budget
Everyone usually thinks of traditional budget elements like the cost of the essential hardware and/or software licensing needed, and consulting costs to assist with implementation. But it's not too hard to overlook other budgetary factors.

Here are a few other potential budgetary considerations to think about to make sure you're planning for the less obvious possibilities that may come along:
- *Maintenance (updates/upgrades):* Many software applications and hardware systems come with a year or several years of access to updates and upgrades (application updates in the case of

software, basic driver and proprietary software updates in the case of new hardware). You want to make sure what you're entitled to as when arrange the initial purchase. Looking at the longer term, you'll want to know what to expect with maintenance renewals for software packages in future years (15 to 20% of license purchase costs is a common annual fee for updates/upgrades, unless your software is cloud based and comes with a consistent licensing fee rather than an upfront purchase).

• **Support:** Is there some sort of support (someone you can call or contact about problems or regarding how to do something) included with the purchase, or not? Many tools and applications will have a menu of support options available, often ranging from some sort of simple support as part of the basic purchase, with paid options for higher end packages providing quicker response times and other advantages.

• **Training:** Have you budgeted enough for training, for initial launch, and training for new staff in the future (see no. 4 below)?

• ***Adequate hardware, bandwidth, and other supporting tech:*** This is a common tech project "gotcha" – something about the hardware, network, or other technology that supports your project is not up to speed to provide an adequate backbone for your new technology, particularly if it is well accepted and adoption exceeds initial expectations. For example, if your technology is web-based, is your school's Internet backbone fast enough to provide consistently acceptable throughput? Are there bandwidth usage peaks that might impact performance at critical times? This issue also falls under the category of "risk assessment" (see #11 below).

• ***Travel & Expenses for consultants:*** This is easily overlooked. If you are working closely with consultants who must travel a fair distance to come to you, make sure you know whether or not you will be charged for travel and expense costs and get some estimates for these for budgeting purposes.

• ***Possible Integrations:*** Do you think you might need to integration this new solution with existing technologies in place in your school? If so, you'll want to try to locate some funds for doing so, or at least consider the possibility.

2. Failing to secure senior administrator/executive level buy-in

To fully position your technology imperative to succeed, you'll want to secure buy-in from some of your school's top administrators or cabinet level managers. Support from one or more upper level administrators can help ensure sufficient resources, and establish your project as a priority. Without it, there's an important support missing from your project's foundation.

Take some time to consider which senior administrators are most likely to want to see your project succeed. Whose departments will benefit the most? If this is primarily an instructional technology that you are implementing, hopefully a Dean or two, or your Chief Academic Officer or Principle, is fully on board, but also consider other areas of the institution that may benefit and see if you can get some of those leaders to back your effort. The ultimate goal is generally getting the senior-most administrator bought in, so reach for the President or Principle if you think you may be able to get them interested in seeing your project succeed.

3. Not having a sponsor

Every project should have a sponsor from outside of the technical department. Someone, preferably in administration, must share ownership for the success of this effort. Your project's sponsor should be from a department that is closely involved with the technology – if it is instructional, a sponsor from

faculty management makes sense; if this a technology related to a specific area of administration, then bring on a sponsor from the applicable areas.

Sponsors should be involved at key points in your project. The ability of a project sponsor to help make sure the project succeeds, and work to resolve potential problem points, is dependent on their role and skills but these are the types of things you may look to your sponsor for assistance with. They don't necessarily have to devote a lot of time to the project (although it is great if they can), but they must be prepared to step up at critical junctures and help the project through trouble spots if needed.

If your project sponsor is someone in your school's senior administration group, you will have achieved two key objectives – sponsorship and senior level buy in – at once!

4. Inadequate training

Another common technology implementation hurdle is providing the right amount of training needed to get users comfortable with the new or updated technology. While there is usually some provision for training, it often isn't really enough. If you want your technology project to succeed, you need to go above and beyond to be sure you can provide enough training to make users productive with it.

If training is costly, then consider offering additional training with in house staff – the "train the trainer" approach. I have often successfully leveraged this technique. One advantage of letting internal staff do the training is that they may have a good sense of how your staff, faculty, and students, may use the technology in question. The more complex and customizable the technology is, the higher the likelihood that how it is used at your school may be rather specific to your institution's policies and processes. Too often, trainers from companies who make these products don't make enough effort to see the tool from your users' perspectives. Your in-house staff can bring this to the training effort while keeping your training costs within budget. For faculty solutions, be sure to look for opportunities to let faculty train faculty.

If finding time for users to commit to training is an issue (and it often is, especially with adjunct faculty), then get creative. Making training resources available in ways that self-starters can leverage can provide another way to meet some training needs and also be available for self-help after initial training. One-on-one training may be hard to fund, but sometimes having some of this available as well (often on a "first come, first served!" basis) can help to fill gaps. You may also want to consider some "lunch and learn" sessions (you should always to try the incentive of some food and refreshments when you train).

5. Insufficient technical testing/validation

It is certainly not unheard of for a tech implementation to struggle or fail because of some undetected incompatibility, or other untested assumption. The technology you are rolling out needs to be tested in real-world conditions, or as near to these as you can get.

The more reflective your testing is of the ways in which the technology will be used when it is in production, the better your chances of uncovering potential issues and addressing them (or at least being aware of them before you "throw the switch"). Be sure to slot time with users to conduct these tests. The most robust testing processes start with developing test scripts. If the new technology is

replacing or updating some existing process, select a handful of random use cases from the way things are currently being done and run them through the new process in a test mode.

A "pilot" is another common approach to testing. Pilot tests can be completely separate from live production use of an application, or they can be true live use in a scaled down environment. In either case, be sure to consider wide ranges of potential uses.

The newer your technology is, the more important it is to gather user feedback in preparation for testing. With technologies that are delivering a lot of new possibilities, users may envision using them in ways that the implementation team is not envisioning. If you can get some of those insights early on, you can consider additional ways in which you may need to test the technology and validate its ability to meet these new functional possibilities.

6. Failure to involve faculty from the onset

Well-chosen faculty members can provide valuable insights into potential challenges, or angles to technology uses that others simply wouldn't think of. This sort of thing is too often overlooked, especially if there is a rush, or implementers are too busy with conflicting priorities - it can be pretty easy to assume that faculty's perspective is already understood based on current uses of existing tools, but that can be a mistake.

Faculty can also be a big help with training (see no. 4 above). If you can get peer training going, you facilitate your project in a number of positive ways.

7. Not understanding the student's perspective

This is similar to the issue above, and the same basic approach can help to circumvent related issues.

I also want to mention another aspect of this that is germane to the educational environment. Often a technology may be focused on one audience, like faculty, but still have a potential impact on the students. For example, a new or upgraded attendance taking or grade management tool - if one of those systems resulted in improper information being stored it would clearly impact students! Make sure you take some time to consider this important angle to your technology project.

8. Assuming 'Relevancy' (will the intended audience want to use it?)

If you build it, will they come? Not necessarily. Of course, this is highly dependent on the type of project – if this is something relatively new, you need to make sure you aren't making an incorrect assumption. This is another reason to capture the perspective of representatives of the intended user community throughout your implementation effort.

Depending on the intended use of the tech, you may also need to do a little selling – how is this used elsewhere, why does it work, and of course – "what's in it for me"? Don't make the assumption that people are going to readily adopt a new technology because "it's cool", "it will help them", or even because management says they have to. If users don't see how it is relevant to their roles, and they don't really like it and can work around it, some of them probably will.

9. Not planning for adequate support

Once the technology is in place, users have to have a functional method for getting questions answered, and resolving issues. Users need to know who to contact and how to contact them, and they need to be able to get their problems addressed in a timely manner. Plaster support contact information all over your efforts – put it out on portals, email it, print it – do whatever it takes to make sure that when a user wants help with the new tool, they know where to find it. Tell them where and how repeatedly.

If this is a new technology for your internal staff to support, make sure that they are given adequate training and all the resources they need to be able to provide the needed support.
Lastly, if you are using some new avenues for support of this new technology, be sure to discuss how you will measure and monitor their effectiveness, and what you can do if support isn't doing a good enough job.

10. Failure to incorporate training into on-boarding for new users

This could easily fall under the category of "limited training" above, but it is really a long term consideration and it often seems to get overlooked. You may put a ton of resources into training during the initial roll out of a new technology, and it may get up and running right on target, but if you don't plan for how new users are going to get up to speed with the tools in the months and years that follow, you may have some challenges with continuing adoption and use of the technology.

11. No/limited risk assessment

One of the elements of project management that is often overlooked is consideration of the assumptions being made and the risks that come from the possibility of some of those assumptions not holding true. At a minimum, there is often an unspoken assumption that many things simply will not change. Can't you just hear it now? "Well, we assumed that …".

It's really not that hard to consider and document some of the assumptions on which the success of your project is based (things like having adequate bandwidth, users liking the solution and feeling they have a need for it, related/dependent technologies being available, and so on). Once documented, circulate the list to stakeholders who can provide perspective on the possibility of any of these assumptions being mistaken or misinformed, then listen to what they have to say and put some thought into what to do if those concerns come to pass. This doesn't have to be a major effort – just take a little time to consider the possibilities so you aren't blindsided if something doesn't work out as intended.

12. Poor (or absent) long term planning

Okay, your technology is up and running, and has worked reasonably well for a year or so now. Time to renew your support and maintenance, or pay the next licensing installment. What's that, you forgot to budget for contract renewal? Uh-oh. Or maybe your technology is so popular and well received that more users than originally intended vie for licensing, or the use of technology consumes more resources (like network storage or Internet bandwidth) than you planned for. Try to make sure you devote a little of your project planning to considering the long term possibilities, to position this new or updated technology for long term success!