Podcast Transcript

In the Classroom 139

Putting SAM to Work in Your Classroom A Rapid Design And Development Model

SEE THE SHOW NOTES AT: *http://tubarksblog.com/itc139*

Stan Skrabut: Well, thanks ever so much for taking time to listen to this podcast. It certainly means a lot. I know you could be doing other things, but you're hanging out with me, and I really do appreciate it. The more you know about instructional design, I am confident, the better you will be able to design your courses. There are so many strategies that you can choose from to include backward design, which we have talked about. There is ADDIE, which I still need to talk about, and SAM. This week, we are going to take a closer look at SAM, or the Successive Approximation Model, as a way of building better courses.

When I was in the Air Force, I had taken courses on instructional systems design or ISD. I was grounded in the proper way to develop a course or other pieces of instruction. It was also during this time that I learned about Gagne's nine events of instruction, which we have talked about previously. As I was pursuing my degrees, I was exposed to ADDIE, which is a waterfall design method that incorporates analysis, design, development, implementation, and evaluation. While this is a solid design method, it is really slow for getting changes back into the design. If done properly, you go through each of these steps one at a time in a linear fashion, and only after you have implemented and evaluated do you go back and make changes to the instruction. You could have had bad instruction early on, but you won't be getting back to it.

Along the way, I also learned about Agile programming and the underlying methodology. Agile programming, it really came from the software world, where you couldn't wait forever to develop a program come to find out it didn't meet your needs, and then you would have to go through this process. It resulted in overbudget of software projects.

During my exploration, I've learned about Agile programming, and I've used that to build a lot of different projects during my time, especially as a webmaster, but I also learned how to apply Agile programming to instructional design. I'm not the only one that saw this connection. Dr. Michael Allen developed a more rapid design methodology called SAM or the Successive Approximation Model. It uses Agile iterative design at its core to test and improve ideas. The goal is to develop working models quicker, you basically prototype a solution, you test it, make updates, and you keep going through this cycle very close to the products that you're making. You get it out there, you build something that's useful, you put it out there, you get some feedback, you make it better, and then you move on to the next part. You will end up

building a useful model and then move on to the next module. You accomplish this through repeating cycles or steps, and so we're going to explore this just a little bit.

The SAM model has three basic main parts, preparation, iterative design, and iterative development. I've included in the show notes a white paper called Foundations of Instructional Design Methodology and Learning Theories, which provides a great explanation of this development cycle that I'm talking about.

Let's take a second to look at each part. The first part is preparation. The first phase is all about collecting information that's going to help your design. You want to get to know your learners, what they do and do not know about the topic. Basically, you're trying to figure out what their prior knowledge is. You're going to take all this information into the iterative design phase. Now, I said that it's iterative, so it's going to go through a number of cycles. In the second cycle of this preparation, you're going to basically be evaluating the results of the first test of your instruction. You're going to see what worked, what didn't, what needs to be changed, and you'll package all that up, and then you move to the second step or phase of the same model.

The second is the iterative design phase. You will either design from scratch or make modifications to the initial design. In the first one, in the first go-round, you're basically going to work from scratch. When it comes around again, you are going to take the input from the preparation phase and figure out where you need to make modifications. The end goal of the iterative design phase is to have goals and objectives developed, determine your evaluation and assessment methods, and sketch out an instructional design. You may have multiple designs initially that you put in front of people and get their feedback, but typically, you'll narrow it down to one design.

Then we move into the next cycle, or the next phase, I guess, is the iterative development. During the first iteration, the first time you're going through this cycle, you are going to develop some mock-ups or prototypes of the instructional materials that you want, and you're going to test them, you're going to get them out there in front of your students, and you are going to be asking for feedback to see if it's achieving the purpose, and you're going to look at the data, and all this goes into play. You really want to see if your instructional activities are going to work.

During the second iteration, you will finalize your instructional materials. This is when you go up and take all that feedback that's come through the design phase, and you're just going to clean everything up. It's a pretty straightforward thing. Then you're going to move on to the next module, and you just going to keep going through this cycle over and over again until you finally have this program or course developed, and everything has been designed, developed, tested, redesign, developed, and basically retested. Really, you should be done by the time you're through the second phase of this because you can sit there and overdesign, and you can spend a lot of time and energy that really doesn't need to be used, but you'll have a really good program or course content or instructional activity at your hands.

This really large design cycle, it's still a formal cycle. It's probably more formal than what you typically use in the classroom. I encourage you to use these principles at the heart of this, where you'll mock up an activity, you'll put it in front of your students, you'll have them go through the activity, but I would also encourage them to provide immediate feedback on that activity so you can make it better for the next group that's going to go through it.

In a more formal model, there's usually a lot more people involved. In the classroom, there's typically just the instructor. In the formal model, designers, they'll build the content, and this is really just for those really large-scale projects. In the classroom, you're going to want to design and develop your activities or modules, and like I said, get them in front of your students. Put a feedback mechanism to capture points where the instruction can be improved. You're going to basically be looking to get feedback throughout the process. In the end, you'll be able to quickly build activities that are beneficial for your students. This is a way that you can keep improving your course.

That is a little bit of the SAM model or Successive Approximation Model. I encourage you to go learn a little bit more, and I put some articles in the show notes where you can get some ideas. It's really not a hard process. You're getting a feel for what your learners need to know. You're drafting out some activities, you're drafting out content, proposals for instruction, and then you build it, and then they use it, and then you improve it. Not a very hard process at all. With that, I'm going to let you go, but here is a quick plug for my book.