Training for Tomorrow

Ce Ce Iandoli, Editor

Three years ago I worked as a faculty intern at a software development company in San Francisco called Macromedia. Macromedia creates software that lots of you may know about: Director, Flash, Shockwave. My job was to walk around in a somnambulant state dreaming of how we could design the home page so that as soon as a user placed her toe on that site, she would be siphoned off to the exact place she needed to go. This was a big deal because there are 18,000 pages sitting under the macromedia.com site and over 3 million people come to it daily.

What tools could I possibly use to answer that technologically-rich question? Colored pencils, colored papers, notes and squiggles, conversations that resulted in no answers, brainstorms and a wild imagination. What was my deliverable? A highly complicated map, resembling a 3-d flowchart, with a psychological quiz for my fictional user. Example, "How do you feel about using computers?" a. Fearful. b. Curious. c. Excited. Really.

Fast forward to another training experience I had just this past fall: I needed to learn web animation software that was especially dense to learn in time for a class about to begin in three weeks. So, I buy 2 books, one containing a CD-ROM with QuickTime movies of how to do things. I insert my CD; play my lessons. Try out the tutorials. And when I get stuck, I look under the ŒHelp, menu. I consult a second book with exercises and pictures which promise me everything will be ok, if only I'd submit to their advice. All of this doesn't do enough for me.

I call a friend who knows Flash (Milissa) and we invite (Sarah) and we meet every Tuesday for months, studying, drinking wine, hating some parts of the program, despising our own slowness at this task, and ultimately, triumphant on two counts: our friendships have deepened alongside our skills. The social/collaborative model worked better for me. When I go to class my teaching assistant who knows more than me (Avery) and my students teach each other tricks and learn the snafus and the quirky fetishes of this new software. Ergo:peer learning.

In this edition of our journal, we watch our authors grapple with new paradigms for training tomorrow's technologists. We make our students think up and down, horizontally and vertically; strategize, doodle, talk with each other; examine how they know what they learned.

We ask them to think about thinking; we insist on self-examination. We create dyads; we listen to what they say as they construct something to catch a phenomenological glance at what matters to them as they work, in teams, in same sex teams, with paper and scissors, and lots of scotch tape.

The tools we need for what comes next are filled with uncertainty. What shall tomorrow's technologists rely on?

We honestly aren't sure. So we rely on our wild imaginations, the logic of critical thinking, and tools that contort everything into 2 and 3-d hopes.

Although we get lost when we stretch to imagine tomorrow's technologists, we need to imagine them as us. And ask something tougher. How shall we talk to them about what we love? And how shall we explain what's lasting and therefore, pretty unimaginable? And how shall we encourage the tension between what's creative and what's feasible?