

05

Coding As Craft: Evolving Standards in Graphic Design Teaching and Practice

Sherry Saunders Freyermuth

Coding As Craft: Evolving Standards in Graphic Design Teaching and Practice

Sherry Saunders Freyermuth

As a professor of graphic design, I am constantly trying to figure out what tools will best prepare my students for future practice in graphic design. Some schools have separate majors for interactive and graphic designers, but as digital media becomes more prevalent in the field of graphic design, can we ever completely separate the two? Can emerging graphic designers afford to be only print and layout specialists? What jobs will be available to them? We need to consider the role of coding as craft as a new pedagogical model in teaching practice. As graphic design evolves, we as professors are tasked with teaching an ever-increasing amount of skills. Design students need to learn the fundamentals of typography, layout, and composition, as well as develop an understanding of the technological tools available today. As if that weren't enough, we'd also like our students to become skilled critical thinkers and problem solvers, so they can navigate this complex field that is always in a state of flux. When I consider my role as an educator, I struggle with how to incorporate coding into more projects.

In a program with just one web design course, I try to incorporate more digital assignments into my graphic design courses, assigning students projects such as creating animated gifs, YouTube skins, and wireframes for apps. I also try to build in discussions and writings reflecting on these changing mediums. These assignments incorporate digital media but do not involve coding. I'm now considering whether or not students should be coding in every class. The organization Code.org, dedicated to including coding in K-12 education, states that by 2020 there will be 1,000,000 more jobs in computing than students equipped to fill those jobs.¹ As graphic design becomes more intertwined with digital platforms, are students going to be pushed out of their own discipline?

At Lamar University, where I currently teach, there is a BFA program in graphic design. Our program has a strong studio focus. Graphic design students must take at least three drawing courses, as well as three foundation design courses in 2D design concepts, 3D design concepts, and color theory. Students must also select upper level studio classes in areas such as photography, printmaking, sculpture, and painting. The majority of graphic design students come to my computer-based courses with a good drawing foundation, an understanding of perspective and light, a strong sense of design and composition, and a good eye for color. However, when students enter my graphic design course, it is as though they are starting over. The skills they have built in previous coursework are directly applicable to their screen-based work, but they have trouble translating them. In the introduction to the book *Becoming a Digital Designer* co-author Steven Heller writes: "Technology has so thoroughly altered the way designers now practice that it is as necessary to be a technologist as it is to be an artist—and many young designers are more involved with the latter than the former, only later fusing technology and art into design."² Heller goes on to say that "The handcraft tradition, so integral to graphic design, has been replaced by pixels. The traditional definitions of graphic design have also been, more or less, tossed aside. And traditional standards of design "beauty" have been usurped by what I'd like to call the digital aesthetic."³ As a designer who values handcraft, I am somewhat nostalgic when I consider what makes a

strong graphic designer. However, I acknowledge that we may be working with an old educational model that values hand over more valuable digital skills.

“Teaching students to become advocates for themselves and their own learning is the key to creating lifelong learners and successful professionals.”

At Lamar University, early course requirements are set by other faculty and administrators, and some of these professional artists may disagree with the importance I place on digital media. Others will tell you that the real computing has to start with your own mind and that the hand is an extension of that. This is a common debate in graphic design education and is well summed up by design educator and theorist, Meredith Davis: “Design educators still ponder the efficacy of pedagogical strategies that introduce students to the computer in their earliest design experiences, rather than focusing on more conventional strategies of making as prerequisites for digital design (such as drawing and wet darkroom photography).”²⁴ Today’s current academic models may not be completely wrong, but as students expand their thinking in graphic design courses, they likely should expand their skills beyond handcraft. My students spend so much time honing their handcraft in other areas of the fine art curriculum, that digital craft and perhaps a beginning model of coding as craft should have a stronger emphasis in graphic design coursework.

Many design educators believe that computer science would be a better area for students to learn code, but unfortunately, there is often a disconnect between the disciplines of computer science and design. Graphic and interactive design educator James Pannafino states that “...students complain that the computer-science teacher doesn’t understand design, or coding was too difficult, and then their portfolios end up lacking any true interactive designs.”²⁵ Pannafino goes on to say the basic markup languages must be taught in design curriculum along

with the principles of interactive design. If a student shows a certain aptitude or interest in more advanced coding and programming such as C++ or Java, professors can guide students to computer science as a minor or secondary area of focus in their coursework. Learning more advanced coding is not a bad idea for students, but it's not a necessary course of action for every graphic designer.

Understanding a Digital World Visually

Coding is a skill that requires practice and repetition. Once students take a course in web design, if they do not build upon and nurture those skills, they frequently have to relearn them when needed in the future. Though most graphic design programs include at least one web design course, effective implementation of coding as practice is still largely absent.

In the classroom, I create and assign projects that involve digital media but do not actually require any coding. I built these assignments based on observations about trends in new media and the job market, and I also reflect on my own professional practice and workflow as I develop these projects. For instance, I have my introductory graphic design class work on a large postcard project (Fig. 1). The whole process takes four to five weeks. Students make four coordinating print postcard designs and go through the entire creative process from conceptualization to print. Afterwards, they must reinvent their design to work as an animated gif. They develop an understanding of the difference in color, pixel resolution, and scale for print versus screen media. Throughout this process, I have students read the Smash Magazine article “The Medium Is the Message,” which reflects on Marshall McLuhan’s theory and relates it to contemporary web design.⁶ Students also take their digital postcards and print and mail them to family or friends. The last steps include discussions of the differences in each experience of using traditional mail versus emailing the postcards, as well as a discussion of changes in media over the past 200 years and the effects on the graphic design profession. The students then write a reflective essay about the project.



Figure 1. Postcard project created by student Melissa Maddox in Fall 2012.

Similarly, I developed an album cover design project for an Illustration class. After a discussion of new marketing techniques, specifically in the music industry, I explained that students would have to create iTunes cover artwork, a Facebook timeline photo and profile image, a YouTube and a Twitter skin for the album (Fig. 2), as well as a physical design that they would be required to print (Fig. 3). In the digital assets, it was interesting to see that the students were very frustrated with the differences in each format. For instance, Twitter content must be left aligned as the browser is fixed on that side. Often a viewer may lose sight of right and left content, and there is a very narrow safe area on the left that will be visible. In contrast, YouTube is center aligned, and all content remains centered as the browser is scaled.

As I was working on this project with students, YouTube unveiled the beta version of their new channels design, which is now very similar to the Facebook timeline photo and profile image.⁷ Viewing these changes firsthand helped impress upon the students the need to stay connected with digital media. Print is still relevant, of course, but students must be made aware of the contrast between the two processes.



Figure 2. Album cover project (digital materials) created by student Thomas Harris in Spring 2013.

In an advanced class focused on branding, I build an understanding of user-centered design by having my class work through the process of wireframing. Students are tasked with creating website and application design mockups (Fig. 4). I have found that my students are extremely capable of thinking through a digital assignment visually. They quickly adapt to screen media, but with any

mention of coding, their comfort level drops drastically. Students' reluctance could be remedied by incorporating coding into more coursework throughout the graphic design curriculum.



Figure 3. Album cover project (print materials) created by student Zachary Scouras in Spring 2013.



Figure 4. Website and application design concepts for individual brand design.

Project created by student Rachel Binagia in Fall 2012.

The Debate

The debate on the topic of how much coding a designer needs to know has been popular in recent years. It is surfacing on blogs and online forums, and there have even been a couple of related posts on the AIGA “Design Educators” website in the past year. Opinions on the topic of how much coding designers must know vary, and, as an educator, I find it difficult to sift through the variety of information provided. Every source has a valid argument and distinct point of view. What I have been able to glean from my research is that although some designers may work only visually (often in Photoshop) on web designs and mockups and then collaborate with developers to finish their projects, most would agree that any graphic designer, even one who wishes to work in print

based media, can benefit by having at least a basic understanding of HTML and CSS markup. These skills are becoming increasingly necessary: thus, I believe they are very helpful for up and coming designers studying today.

There are other pedagogical techniques that are used to provide web skills to students. As I have mentioned earlier, in discussing my classroom projects, coursework can focus on the visual aspects of designing for a digital space. The combination helps students understand the requirements of designing for screen media such as 72 dpi resolution and RGB color. Students can also work through design problems involving interactivity and acquire skills such as prototyping, user experience design, user-related research, wireframing, and web planning before ever thinking about code. Paper prototyping and early user testing are great ways for students to understand the effectiveness of their web designs before creating website mockups or toiling over code. Author and web developer Jen Kramer agrees that creating visual mockups is a good way for students to get started in web design: “It enables students to picture a Web page in their minds without being too concerned about how the page would be coded.”⁸ Kramer adds some other ways to introduce web topics without focusing on coding; including getting students to think beyond the canvas in which they are designing and think about topics such as responsive design, touch-based media, and grids, while generally working in a flexible environment. These are critical thinking skills that will be of great help as a student transitions into the next phase of web education and begins to code.



Print is still relevant, of course, but students must be made aware of the contrast between print and digital.



I believe there are opportunities to build projects that involve HTML and CSS markup in courses outside of a web design class; it just may take some tweaking and restructuring of current coursework. For instance, I am currently in the

process of developing an HTML/CSS focused project for a typography course. Inspired by the “Don’t Fear the Internet” tutorials by Jessica Hische and Russ Maschmeyer, students will be required to make an online menu for a restaurant of their choice.⁹ Students will create their work locally on their own hard drive and focus on creating a one-page web layout using HTML and CSS. The focus and singularity of the project make it a great way to begin understanding the basics of both coding and typography for web platforms. It works well for both introductory students and those who have taken classes in web design—as even more advanced students can use the additional practice. The final portion of the project is to create a corresponding print layout of the menu students have created digitally. By manipulating assignments to include web and coding skills, educators can maintain quality graphic design instruction while incorporating the digital education many programs lack.

The second argument in this debate is whether or not a technological focus in graphic design education will detract from creativity. This is less about markup languages such as CSS and HTML and more about in-depth programming and web development. Some argue that designers need to be skilled in more complex web languages such as PHP and Javascript in order to create achievable designs that developers can actually make. Having such a skill also frees designers from having to work with developers and makes them more marketable as freelance designers. On the other hand, there is the argument that too much focus on the limitations of code keeps designers from more innovative and creative work. In interaction design, there are always some designers who are technologically focused and others who are user-experience focused. Interaction design consultant Christopher Fahey summarizes the issue succinctly:

Yes, some designers will delve deep into technology, being hands-on coders and fabricators of interactive artifacts. In fact, some great interaction designers already spend most of their days thinking of themselves primarily as technologists. Others, however, will focus on the design parts of interaction design. These people will most often

work closely with other individuals and teams to implement their designs. In short, great design will come from great designers, and great technologists will make those designs happen. Sometimes these skills will be found the same person, but increasingly not. An interaction design education should support both models, of course.¹⁰

I agree with Fahey that an interaction design program must support both models, but as graphic design educators, we often have other considerations in our curriculum, and I would argue that advanced coding and web development skills can be left to the student to decide how much they need. There are many great tools and websites available, such as Lynda.com, Teamtreehouse.com and Codeschool.com and many others, that can help anyone who desires to learn advanced code on their own.



Will the software catch up to meet our needs?

I like to believe that the marketplace is catching up to the needs of designers and is developing the tools necessary for visually creating web designs. Adobe Creative Cloud, for example, has many promising tools such as Muse, Edge, Game Development Tools, and the Digital Publishing Suite. Also, content management template-based sites such as Wordpress, Jumla and Drupal make the web design process much more manageable for designers. They are not code free, however, and do require some knowledge of coding in order to have complete control. These tools seem promising, however, this 2012 statement from web designer Jake Rocheleau indicates we are still far away from purely visual web design: “In an ideal world we would be able to build graphics and easily import them into a program for compiling into a website, desktop app, whatever. Sadly this is not the case and doesn’t appear to be anywhere on the horizon. For now web designers and developers must work together to build projects which are stunning visually and function appropriately.”¹¹ For graphic designers to remain empowered on the web, they must at least keep up with

trends in web design and digital media, as well as have a good understanding of HTML and CSS—with any additional coding knowledge as a helpful knowledge base.

“Educators need not compromise the fundamentals of design to develop more technologically advanced graphic designers. A balance of hand and digital craft must exist.”

The reason for considering the addition of more coding in my graphic design curriculum has everything to do with the current trends in the job market. With digital media becoming so prevalent in the lives of the modern consumers or users, companies are increasingly looking for designers with digital and web experience, in addition to the fundamental skills of layout, typography, and conceptual design thinking.

Take this condensed description of a recent job posting for a junior designer on Aiga.org:

Junior Designer at Deutsch Family Wine & Spirits

White Plains, New York

Job Level: Junior Staff

Qualifications: Superior knowledge of design: type/color/taste, Mastery of relevant computer software (e.g. Adobe Creative Suite); solid knowledge of other current computer design programs and skills, solid understanding of online and offline production, web skills with CSS, HTML, UI/UX and video editing experience preferred.

Posted August 2, 2013.

These are the types of jobs our recent graduates will be applying for as they graduate from our design programs. From the list of qualifications, which are commonly found on entry-level and often in-house design positions, it is evident that employers expect students to be jacks-of-all-trades and understand everything in the print and digital realm, which, to many employers, includes web design and coding.

Perplexed over whether or not this list was an achievable amount of skills for a junior designer, I came across a very telling job posting, which was actively looking for a “designicorn.”

Designicorn at Switch Studio

Venice, California

Job Functions: Account Management, Administrative & Support, Art Direction, Branding & Corporate Identity, Creative Direction, Design, Design for Social Impact, Information Architecture, Media, Non-Profit, Packaging, Project Management, Publishing, Strategy & Planning, User Experience, Web Design, Web Development
Description: Switch Studio is expanding and we’re looking to find that mythical (but totally real) über-designer who is equipped with a mix of future-skills, an attention-to- detail personality, and a can-do mindset. Posted July 24, 2013.

Although witty and fun, the description speaks to a pressing problem in the job market— specialization could potentially limit your career prospects.

In this difficult economy with enormous competition for design positions, graduates are daunted by the expectations of the job market. As an educator, I want to help develop the confidence of my students and make sure they are lifelong learners that can get any type of job they are after, and I believe a better understanding of coding and user-experience design is important to incorporate throughout the design curriculum.

Staying relevant?

How are faculty members going to stay relevant in this fast-paced and evolving world of coding? I attribute this process to learning an instrument. It requires time, effort, and consistent practice. Kramer very accurately describes my own situation when addressing the common problem of full-time faculty keeping up to date: “Full-time instructors and professors typically don’t build websites professionally (or they’ll build few of them), and with many other demands on their time, they may have a hard time keeping up with trends and techniques.”¹² As a full-time graphic design educator, one can feel split in so many directions that it can be quite a challenge to prioritize advanced web techniques in day-to-day work. One helpful suggestion by Kramer, in addition to collaborating with professional web developers, is to help students learn to learn:

Students can’t always expect a teacher to spoon-feed them what they need to know next. They need to figure out how to learn without courses or books in order to stay on the cutting edge. Help students curate interesting blogs and social-media resources for finding tips and techniques.¹³

Teaching students to become advocates for themselves and their own learning is the key to creating lifelong learners and successful professionals.

Conclusion

As I consider how the new generation of design educators will prepare emerging students for these complex and evolving changes, I look to the epilogue of *Meggs’ History of Graphic Design*: “As so often in the past, the tools of design are changing with the advance of technology. The essence of graphic design, however, remains unchanged. That essence is to give order to information,

form to ideas, and expression and feeling to artifacts that document human experience.”¹⁴ Educators need not compromise the fundamentals of design to develop more technologically advanced graphic designers. A balance of hand and digital craft must exist. The solution, then, is exactly that; maintain our fundamentals while understanding the changes and needs of the discipline as it evolves in this new era.

Endnotes

1. “What’s Wrong with this Picture?” Code.org, <http://www.code.org/stats>.
2. Steven Heller and David Womack, *Becoming a Digital Designer: A Guide to Careers in Web, Video, Broadcast, Game, and Animation Design* (Hoboken, N.J.: John Wiley & Sons, 2008), 12.
3. Heller and Womack, *Becoming a Digital Designer*, 17.
4. Meredith Davis, *Graphic Design Theory* (London: Thames & Hudson, 2012), 213.
5. Pannafino James, “Learn That Over There... Do Design Students Need to Learn to Code Within a Design Curriculum?,” *AIGA Design Educators Community*, <http://educators.aiga.org/learn-that-over-there-do-design-students-need-to-learn-to-code-within-a-design-curriculum/>.
6. Jason Gross, “The Medium Is the Message,” *Smashing Magazine*, <http://www.smashingmagazine.com/2011/07/04/the-medium-is-the-message/>.
7. Justine Ezarik, “YouTube Creator Blog: How to Prepare for the New Channels Design!,” *YouTube Creator Blog*, <http://youtubecreator.blogspot.com/2013/02/how-to-prepare-for-new-channels-design.html/>.
8. Jen Kramer, “Teaching Web Design to New Students in Higher Education,” *Smashing Magazine*, <http://www.smashingmagazine.com/2013/08/05/teaching-web-design-to-new-students/>.
9. Jessica Hische and Russ Maschmeyer, “Don’t Fear the Internet: Basic HTML & CSS for Non-Web Designers,” *Don’t Fear the Internet*, <http://www.dontfeartheinternet.com/>.
10. Christopher Fahey, “Are We Designing Interactions or Designing Software?,” *Graphpaper*, http://www.graphpaper.com/2009/02-11_are-we-designing-interactions-or-designing-software.
11. Jake Rocheleau, “How Much Code Should Web Designers Need to Know?,” *Speckyboy Design Magazine*, <https://speckyboy.com/2012/03/22/how-much-code-should-web-designers-need-to-know/>.
12. Kramer, “Teaching Web Design to New Students in Higher Education.”
13. Ibid.
14. Philip B. Meggs and Alston W. Purvis, *Meggs’ History of Graphic Design* (Hoboken, N.J.: John Wiley & Sons, 2012), 572.

Author Affiliations

Sherry Saunders Freyermuth

Assistant Professor of Graphic Design, Lamar University.

A good foundation in graphic design history, theory, and practical application will help you hit the ground...Â Graphic designers solve visual problems. The key to teaching yourself graphic design is to understand the process of solving a visual problem. This means you'll benefit from tackling design briefs. You'll learn to apply the skills you study by solving fictitious design problems to begin with and as you advance tackling real world problems and working with clients.Â These type of practical and technical limitations are an important part of practicing the craft of graphic design. Work to learn about these technologies and build up your knowledge through real projects. You'll learn a whole lot about print by having to get a big project printed on a budget.