Artifacts in End-User Web Design

Thomas H. Park  
College of Information Science and Technology  
Drexel University  
Philadelphia, PA 19104 USA  
thomas.park@ischool.drexel.edu

Susan Wiedenbeck  
College of Information Science and Technology  
Drexel University  
Philadelphia, PA 19104 USA  
susan.wiedenbeck@ischool.drexel.edu

Abstract  
In this paper, we contrast the artifacts that professional and end-user web developers utilize when designing a website’s interface. We then briefly discuss how these differences hold implications in effective tool support for end-user web developers.

Keywords  
End-User Development, Web Design, Artifacts

ACM Classification Keywords  
H.5.2. Information Interfaces and Presentation: User Interfaces: User-Centered Design

Introduction  
A significant proportion of people who build websites are end users who engage in web development for personal, recreational, and work-related reasons. For instance, the task of building a small library’s website is often undertaken by a librarian rather than a professional web developer.

These end-user web developers differ from their professional counterparts in a number of ways. Although they are also concerned with the design of their websites, it is the content that primarily motivates their activity. They often lack the training, experience, and expertise with which professionals are equipped. End-user web development tends to be opportunistic,
shaped broadly by organizational constraints such as time and budget [3].

Artifacts feature prominently in the web development processes of both professionals and end users. In our ongoing research, we are exploring key differences in how these two groups interact with artifacts. The focus of this paper is on representations that are used in the design of a website’s interface – that is, the visual organization of navigation and content elements.

**Wireframes, Mock-ups, and Prototypes**
Professional web development is "a process of iterative refinement that moves the design from high-level and general to increasingly specific and detailed" through the use of intermediate artifacts [2]. Professionals initially explore design ideas with pen-and-paper sketches and later use software tools to refine the ideas with promise. Wireframes, which can have varying degrees of refinement, are used to visualize a webpage’s structure while keeping the content ambiguous (figure 1). Mock-ups are static images that incorporate graphics and content to provide high-fidelity representations of websites. Prototypes are used to demonstrate interactive behavior and test the user experience of websites.

In experimental settings, end-user web developers also value the creation of artifacts for design planning, with "informal sketches of possible screens [being] a ‘natural’ approach...” [5]. Nevertheless, we have found that in practice, end-user web developers rarely turn to wireframe sketches and other low-fidelity artifacts to support their design activities.

![Figure 1. A wireframe sketch](image)

In professional web development, intermediate artifacts are used to communicate intent, assess progress, and develop common understanding among designers, developers, managers, and clients. This collaboration is full of ambiguity, banter, and continual negotiation [1]. In contrast, end-user web developers act as webmasters, individually responsible for most aspects of their websites. Their communication usually consists of gathering requirements early and collecting feedback later from non-web developers.

**The Website Itself**
The website itself serves as the central artifact with which end-user web developers engage, mediated through an integrated development environment (IDE), text editor, or other web design program. Designing through the manipulation of a website itself is beneficial because of the website’s mutability, immediacy and
fidelity of feedback, and communicability. Yet with this approach, end-user web developers often grapple with multiple facets of development at once, from high-level interface design to graphic design, content editing, and coding. Because they often lack expertise in several of these activities, engaging in them simultaneously can compound the challenges that they face.

In response to these challenges, end-user web developers adopt a “functionality first” approach that ensures a working website but limits the breadth of design. Once a webpage has taken shape and is functioning as desired, they advance the design incrementally through minute cycles of reflection-in-action [6]. In these cycles, they make a small change, preview it in the browser, and then save or undo it. When major changes are required, they save a backup of the page before proceeding, using various file-naming schemes to improvise versioning systems.

This “functionality first” approach suggests another reason why end-user web developers may not utilize the intermediate artifacts of professionals. They do not know the precise limits of their abilities and lack the confidence or ability to make the leap from unfettered design sketches to implementation in code.

**Templates and Widgets**

A common way that end-user web developers reconcile their “functionality first” strategy with their desire for appealing designs is through the use of website templates. In lieu of sketches, an end-user web developer may browse templates to weigh potential layouts (figure 2). Once a template is selected, it serves as the skeleton for their website. Because templates are self-contained, they provide some assurance of their functionality while offering designs that are relatively sophisticated. End-user web developers similarly rely on widgets for advanced functionality such as multimedia and interactivity (figure 3).

![Figure 2. A gallery of templates (www.freecsstemplates.org)](image)

Problems do occasionally arise when end-user web developers implement or modify a template or widget’s prefabricated code. For example, replacing a template’s filler text with their own content can cause rendering issues that must be debugged. In these cases, they can experience considerable difficulty in making sense of unfamiliar code.

As a result, end-user web developers rarely reuse their own or others’ code [4] apart from templates and widgets. More frequently, they scour other websites for
design ideas, with attention paid to layouts, images, and color schemes [3].

![Figure 3. A virtual bookshelf widget (www.shelfari.com)](image)

**Discussion**

Professional and end-user web developers contrast starkly in their use of artifacts and artifact-centric strategies. Therefore, effective tool support for end-user web developers requires different models and representations than are used in professional tools. They should be based on an understanding of end-user web developers’ practices, the limitations of them, and the rationales behind them.

For example, “functionality first” permeates end-user web development and explains the use of templates. But given the emergent nature of design, choosing a template often leads to committing prematurely to a layout. Moreover, templates can cause end-user web developers to conflate multiple facets of web development, including layout, color scheme, and typography. Tools that tease apart these facets without abandoning templates as representations can assist end-user web developers in ensuring functionality while gaining greater flexibility in their designs.

On a final note, while end-user web developers are concerned with getting their content online, they are also driven with personal interest in web development. Thus, tools should be designed to foster their motivation, self-efficacy, and ability so that they might grow as web developers.

**Acknowledgements**

This work is supported in part by the EUSES Consortium via NSF grant CCR-0324844.

**References**


