

U.C. Berkeley Web Design Patterns Library

May 4, 2006

Tim Dennis, David Hong, Mano Marks, and Kelly Snow
School of Information, University of California, Berkeley

We thank Owen Otto for his contributions to this project. We particularly thank our advisor and Director of Center for Document Engineering, Bob Glushko, for his guidance on this paper and our project in general.

Introduction

Our primary motivation for this project was to consider why user experience, particularly around web-based user interfaces, is so poor across the Berkeley domain. Why is a good user interface so difficult to create? Why is it that at U.C. Berkeley, a leader in technology research, a place of high academic achievement and innovation, good web user experience is hard to come by? To answer these questions, we conducted a survey and follow-up interviews, detailed in the reports section at the end of this report.

The U.C. Berkeley environment

Based on survey user interviews, we found that the environment for U.C. Berkeley web developers has the following characteristics:

- Heterogeneous development environments in terms of
 - Code based used (JSP, PHP, Static XHTML, Content Management Systems, CGI, etc.)
 - Size of design team
 - Experience and training of web developers
- Balkanized department administration leads to duplication of effort, lack of central control, and lack of responsibility for decisions.
- Resistance to central authority. Developers often stated that either they or other developers they knew were resistant to any mandates or resources provided by the central campus administration.
- A focus on database development.
- Little incentive or resources to evaluate or improve user experience.
- Individual developers feel isolated and unsupported on their projects.
- Successful cross-campus applications started bottom-up. One developer or development team created something and shared it with others. Eventually, it would get more and more widely implemented until it became a standard.

We observed this is not just a U.C. Berkeley problem; it's common at many universities. Sakai, a cross-university, open source community software development effort to design, build, and deploy a course management and learning environment has similar characteristics. ¹

Solutions Considered

We spent a semester researching various solutions to these problems.

Code Library

Several of our interviewees suggested a code library as a solution. We rejected this solution for the following reasons:

- 1) Code would not solve user experience problems. The fundamental issues we encountered were layout and navigation issues. There was no way to develop code to work in all situations.
- 2) The heterogeneous nature of development environments on campus made it unlikely we could capture all developer needs.
- 3) Code changes rapidly, meaning that without constant maintenance, a code library would be rapidly out of date.

Model-based Applications

Model-based applications follow an approach that creates an interface from conceptual models of an application. This approach largely rests on the notion that in order to facilitate a flexible development process, a developer would create an abstract interaction model using an encoding technology, usually an XML markup language. This model document is then transformed to target interfaces using rules also encoded in an XML document. We considered some existing and emerging technologies such as Mozilla's XUL (XML User Interface Language) and Microsoft's XAML (Extensible Application Markup Language). While both technologies provided some semantics to describe an interface, we felt they would not work for our user base for the following reasons:

- 1) They are not technology-neutral and tied to specific runtime environments not available to all users.
- 2) They require a great deal of specific technical knowledge as well as design knowledge. This would limit people using it with existing applications.
- 3) Resistance to central mandates on campus would probably stop their implementation.

Graceful degradation and Other Rule-based Automation Techniques

A principle that a user interface should be flexible in such a way that if another media cannot cope with some of the elements on the screen, it can still be displayed in a simplified form.² For example, mobile phone web browsers must deal with highly interactive web pages in a different way given the form factor and small screen resolution. While this topic seems to veer off the subject of design patterns, graceful degradation speaks to a bigger problem of maintaining consistency in user interfaces over a variety of client interfaces. However, graceful degradation did not address the needs of our users at all. In addition, these techniques are implemented when there are disparate devices implementing the browser. We found that our target users were, for the most part, not even considering support of non-computer based display.

Web Design Patterns Solution

Ultimately we decided Web design patterns were the right solution for our audience.

- 1) Web design patterns capture recurring problems and potential solutions in user interfaces. Though Web design patterns do not provide explicit implementations, they are intended to help designers and appropriate stakeholders make decisions that would help users achieve their end goals. This was the right solution for our environment for the following reasons: It is not a central mandate on technology or implementation. Rather it is a codification of best practices and suggestions for use.
- 2) Our library allows for user feedback and user-developed patterns. This gives our target users a chance to become a community of designers, sharing knowledge and resources.

History of UI Design Patterns

While design patterns have existed for some time in architecture and in software development, user interface design patterns have only recently begun to attract a large number of users. We believe that design patterns, specifically UI design patterns, can help mediate the often contentious process of developing user interfaces.

Knowing that development environments in the open-source community as well as the university campus are often scattered, patterns can provide a solution for social

knowledge management. As such, the development of patterns and a pattern library addresses the need to bring together these shared interests and to engage in some mutual enterprise.

What We Accomplished

Ultimately, we built a web-based application that contains web design patterns. But there is more to this story than simply building an application:

- This project incorporates concepts from organizational psychology to develop a community-based, collaborative tool.
- User-centered design methods were followed to design the interface and determine features to support our users.
- We developed and executed a technical strategy that will easily integrate into the existing university environment.
- The application was technically implemented with best practices; taking advantage of code reuse practices, and abstracting presentation and database layers.
- Patterns were developed from a document model and in the process of doing this, developed a new methodology for creating patterns.

How We Did It

After conducting a survey of the campus web development environment,³ we then identified common problems that we saw across some campus applications.⁴ We found that campus applications in general had four main issues:

1. Inconsistent user experience on sites;
2. the lack of location indicators and clear navigational schemas;
3. the need to maximize content areas for forms and other dynamic data;
4. and finally the need to display student information.

With these ideas in mind, we harvested various patterns and honed our efforts on developing 21 patterns as part of our library of patterns to use.

We also needed to understand how to organize a pattern library in a way that best supports campus web designers. We conducted a comparative analysis of the major Web design pattern collections: Yahoo! User Interface Patterns, Jenifer Tidwell's *Designing Interfaces*, Martijn van Welie's *Web Patterns*, and van Duyne, et.al.'s *Design of Sites*.⁵ From this comparative analysis, we created a methodology for selecting and writing Web design patterns.⁶

Finally, we took suggestions from our initial survey, follow-up interviews⁷, and user tests.⁸ From our survey, we found that PHP was the most commonly used language on campus and subsequently, took the suggestion to use PHP.net as a familiar model for presenting information along with the ability to allow users to comment on patterns. The commenting feature is one of the unique characteristics in the development of our repository when compared with the others. Users can also submit and contribute to the actual patterns as well.

Future Direction

We believe that patterns provide a model for common problems and appropriate solutions in highly diverse Web design development environment. However, for Web

design patterns to flourish in such an environment, there is a need to support an information ecology around which the patterns are created and subsequently used. To this end, we believe the university needs to allow people to engage in activities where ideas can be collaboratively created and shared. Currently, the Web development environment on campus is not conducive to such an environment. In fact, we have found contrary conditions where:

- knowledge about user interface and design is typically not distributed or shared
- knowledge about what constitutes good user interface is inconsistent
- each person has their own ideas and agendas about the interface

We believe this condition can find its corrective in community-based applications, such as our, which provide a mechanism and toolsets that developers can use to begin the task of developing communities and building up common languages around design issues. We believe in this way disparate development environments can better attain consensus on something as contentious as the user interface. More importantly, possibly, the campus can start to provide richer and more consistent experiences for its user communities.

The Program Office at U.C. Berkeley agrees with us. This project is in the process of transitioning into that office for further development and study. In short, our project will be available to our users and have ongoing support.

Reports

[User Interface Design Patterns: Strengths, Challenges and Future of Design Patterns](http://ui-designpatterns.org/tr/uidp_strengths_challenges_future.pdf)

(http://ui-designpatterns.org/tr/uidp_strengths_challenges_future.pdf)

This paper examines the emerging field of user interface (UI) design patterns. We will look at what design patterns are, how they can help the design process, their limitations, and what we see for the future of user interface design patterns.

[U.C. Berkeley Web Designer & Developer Survey 2006](http://ui-designpatterns.org/tr/UCBWebDevSurvey.pdf)

(<http://ui-designpatterns.org/tr/UCBWebDevSurvey.pdf>)

The purpose of this survey is to understand computing and programming employees at U.C. Berkeley. Specifically, it aims to understand the characteristics of Web application and Web site designers and developers. We investigate the technical skills of this population, where these employees are allocated across campus departments, their job function, and what resources they access to support Web development and design.

[User Interviews Summary Report](http://ui-designpatterns.org/tr/UserInterviews.pdf)

(<http://ui-designpatterns.org/tr/UserInterviews.pdf>)

We conducted 16 user interviews to understand our potential users, determine specific target users, and to determine a strategy for developing a web design pattern library. We conducted two rounds of interviews to focus on a specific user type, the Swiss Army Knife.

[Analysis of U.C. Berkeley Web Applications for Web Design Patterns](http://ui-designpatterns.org/tr/ComparativeAnalysisOfCollections.pdf)

(<http://ui-designpatterns.org/tr/ComparativeAnalysisOfCollections.pdf>)

The goal of this analysis is to understand Web design patterns currently in use in U.C. Berkeley Web-based applications. This analysis identifies opportunities to implement patterns not currently in use or not currently developed.

[Comparative Analysis of Web Design Patterns & Pattern Collections](http://ui-designpatterns.org/tr/ReviewOfCampusWebApps.pdf)

(<http://ui-designpatterns.org/tr/ReviewOfCampusWebApps.pdf>)

The goal of this comparative analysis is to understand current approaches to organizing and displaying a user interface pattern library and to understand if there are any standards in the way patterns are displayed or written. Additionally, we wanted to determine if there are similar or different patterns across collections.

[Methodology for Developing Web Design Patterns](http://ui-designpatterns.org/tr/MethodologyPaper.pdf)

(<http://ui-designpatterns.org/tr/MethodologyPaper.pdf>)

This paper outlines a methodology for developing Web Design Patterns and selecting patterns for a pattern library. We developed this methodology in order to create the U.C. Berkeley Web Design Pattern Library.

[Designing and Testing an Interface for the U.C. Berkeley Web Design Pattern Library](http://www.ui-designpatterns.org/tr/usertesting.pdf)

(<http://www.ui-designpatterns.org/tr/usertesting.pdf>)

This paper describes the user-centered design process and methods used to develop the web design patterns library application.

[Feature Requirements](http://ui-designpatterns.org/tr/FeatureRequirements.pdf)

(<http://ui-designpatterns.org/tr/FeatureRequirements.pdf>)

This document specifies the features to be implemented in the U.C. Berkeley Web Design Pattern Library.

[Technical Design](http://www.ui-designpatterns.org/tr/techdesign.pdf)

(<http://www.ui-designpatterns.org/tr/techdesign.pdf>)

This document outlines the technical design of the web patterns repository.

References

¹ Sakai - <http://www.sakaiproject.org/>

² Florins, Murielle and Jean Vanderdonckt, "Graceful Degradation Of User Interfaces As A Design Method For Multiplatform Systems." International Conference on Intelligent User Interfaces, Funchal, Madeira, Portugal, January 13 - 16, 2004.

³ Snow, Kelly and Marks, Mano, "U.C. Berkeley Web Designer & Developer Survey," Document Engineering, Tech Report CDE2006-TR02, January 2006.

⁴ Dennis, Tim and Snow, Kelly, "A Comparative Analysis of User Interface Pattern Collections," Center for Document Engineering, Tech Report CDE2006-TR04, April 2006.

⁵ Snow, Kelly, and Hong, David, "A Comparative Analysis of Campus Web Applications," Center for Document Engineering, Technical Report CDE2006-TR05, April 2006.

⁶ Marks, Mano and Snow, Kelly, "Methodology for Developing Web Design Patterns," Center for Document Engineering, Technical Report CDE2006-TR06, April 2006.

⁷ Dennis, Tim and Snow, Kelly, "User Interviews Summary Report," Center for Document Engineering, Technical Report CDE2006-TR03, April 2006.

⁸ Dennis, Tim, and Hong, David, "Designing and Testing an Interface for the U.C. Berkeley Web Design Pattern Library", Center for Document Engineering, Technical Report CDE2006-TR07, April 2006.