

# LIS901-03

## User interfaces

2009-07-24

See the course web site at <http://openlib.org/home/krichel/courses/lis9013n09a> for the latest online version of this file.

### Course Description

This course extends LIS650 to build interactive web user interfaces. While there user interfaces are very general, the course uses examples that are most frequent in library scenarios. These concern menus (for content-rich library sites) as well as help features when filling out forms.

Generally, a user interface on the web is provided by a browser. Thus a browser defines what users can do with web pages. The way users interact with the pages as written in LIS650 is limited. Users either move from page to page following links. Or they use a browser feature that is common to all pages. Such features include bookmarking a page, going back to a previous page, manipulating the address bar etc. As a rule, in the sites constructed in LIS650, there is no way that the web page itself changes as a function of user actions.

Modern pages have overcome this limitation. They include code that make changes to the page as the user uses it. This allows designers to build pages that interact with the users. In this course we aim to build such pages. Many interactions are simple and fairly standard. For example, a pull-down menu helps to preserve space on a page. Or, for another example, a page may open a new browser window when a link is followed. Or, for yet another example, as users roll over a sequence of small thumbnail pictures, each of them, in turn, is shown in a large format in another part of the page. One of the most useful moments for interactive pages is when we are collecting input data from a user. Users enter data using buttons and forms. When users fill in a form, we can help them using interactive features. We can display hints. We can check the input. And we can warn users if they have submitted values that must be incorrect.

Interactive pages can also be used for purely artistic purposes. As users navigate a page, it can be changed around completely. But this is not what the course intends to focus on. Instead we will try to limit page interaction to features that help the users to navigate the site, to find the right page, and to understand the contents the pages. In course of doing that, we also investigate what distinguishes a good interactive page from a lousy one.

The course concentrates on user interfaces as an application area. But the material covered includes two extremely important techniques for digital librarianship in general. These are the Document Object Model (DOM), and regular expressions, respectively.

The DOM is a standard for the manipulation of contents in XML. As we have seen in LIS650, web pages are written in XML as HTML. Thus changing them as a response to user action involves changing the information on the page as encoded in XML. In this course we change XML as a result of user actions. But the DOM can also be very useful in massaging massive amounts of XML data. While such use is not part of the course, it is one of the most frequent digital librarianship tasks.

Regular expressions are a way to express pieces of information. For example, a phone number has 10 digits. It may be written as three digits, a space, then four digits, then a space and then three digits. Or it may be written as three digits, a dash, then four digits, then a dash and then three digits. A regular expression can be used to find out if the phone number has the proper format.

The course applies these two standard technologies within the confines of JavaScript. JavaScript is the standard language used to add interaction to web pages. The course will only examine a part of this language. That part is the basic usage of the DOM within JavaScript. The course is not meant to cover JavaScript in great depth.

In addition, the course covers some issues of interface design. How can we judge a user interface? How can we distinguish a good interface from a bad interface?

### Course objectives

After taking this course students

- will understand fundamental concepts of user interfaces
- will have basic command of JavaScript
- will be able to use the DOM for manipulation of XML contents

- will be able to use the interplay between JavaScript and CSS
- will have a basic grounding in the usability of interfaces
- will be able to judge typical library web interfaces from a usability point of view

### Prerequisites

Since the practical work is based on standard compliant XHTML 1.0 and CSS level 2.1, students have to have taken LIS650 or demonstrate equivalent expertise.

### Instructor

Thomas Krichel

Palmer School of Library and Information Science  
 C.W. Post Campus of Long Island University  
 krichel@openlib.org  
 work phone: +1-(516)299-2843  
 skype: thomaskrichel

Private contact details may be obtained from the online CV at /home/krichel/cv.html.

### Class structure

Classes are held in the PC1 lab of Bobst Library on Saturdays between 13:00 and 18:00. The instructor promises to be in room 707 shortly after 12:00 for extra help and questions. After class the instructor will lead the students to the local bars.

Each class has a presentation by the instructor. For some part of class time the students work directly with their computers under the supervision of the instructor. Students are expected to do some independent work at home. To support the students in this process, the instructor will be on campus in extra sessions for students who need additional support. Support via skype is available pretty much around the clock, i.e. unless the instructor is moving around or is asleep.

Class details:

2009-10-24	13:00 to 18:00	browser interfaces, scripts, and events
2009-10-31	13:00 to 18:00	basic Javascript objects
2009-11-07	13:00 to 18:00	Javascript DOM
2009-11-14	13:00 to 18:00	form and regular expressions
2009-11-21	13:00 to 18:00	more on regular expressions
2009-12-05	13:00 to 18:00	conclusions

Slides for all classes are downloadable from the course web site. The slides on the course website are drafts until the time that the class is held.

### Tools

A critical tool that belongs on the "desk" of everyone working with the web is the FireBug extension of the popular FireFox browser.

Weyl (2007) has written an introduction.

### Readings

Flanagan (2006) has the classic title on JavaScript. But we more rely on the DOM scripting approach as introduced in Keith (2007).

?) is a classic book that has a lot of entertainment value.

In other bits and pieces, Nielsen (2009) praises large drop-down menus for accessibility.

Finally there a bunch of home-grown resources <http://openlib.org/home/krichel/courses/lis653>.

### Mailing list

There is a mailing list for the course at <https://lists-1.liu.edu/mailman/listinfo/cwp-lis653-krichel>. All students are encouraged to subscribe. As a rule, answers to email sent to the instructor are copied to the list. There are exceptions to this rule

- if the question writer requests the answer not to be posted;

- if the question is a purely private matter.

### **Assessment**

Before each class except the first, there is a quiz on the issues covered in the previous class. The average of all the quiz results counts for 5/12 of the assessment. From the third class onwards, students will prepare a one web-page homework that illustrates the principles covered in class. At the beginning of each class when homework is due, students assess each other students homework and rank the pages. The sum of these is worth 4/12. The ranking as then used to grade the pages done. The final project is a single page illustration a form entry problem with hints and value checking. It must be well documented inside. The final page is worth 3/12.

### **References**

- Flanagan, David (2006). *JavaScript: The Definitive Guide* (5th ed.). O'Reilly. see <http://oreilly.com/catalog/9780596101992/>.
- Keith, Jeremy (2007). *DOM Scripting: Web Design with JavaScript and the Document Object Model*. Friends of Ed.
- Nielsen, Jakob (2009, 23 March). Mega Drop-Down Navigation Menus Work Well. Alterbox.
- Weyl, Estelle (2007). Introduction to Firebug. available at <http://www.evotech.net/blog/2007/06/introduction-to-firebug/>.