PUDT 2101 D / CRN: 4382
Thursdays 7–9:40pm / 55 W 13th, Room 404

Core Lab: Interaction!

Spring 2013 Syllabus

Taught by: Lev Kanter
kanterl@newschool.edu
646 641 5329

TA: Catalina Cortázar
cortc105@newschool.edu

About the Course

The official course description, provided by the department, for *Core Lab: Interaction* is:

> *This core lab provides hands-on production skills and processes for projects in Core Studio: Interaction. Students will learn processes and techniques for website and interactive design, media integration and problem solving. In-class projects will be complimented by the production of a rich-media interactive portfolio. Primary technologies include: HTML, CSS, Javascript, and jQuery.*

Core Lab: Interaction is about creating user interfaces, and understanding how they work. The main focus will be on making things for the web. We will learn and practice the fundamentals of how to write HTML, CSS, and Javascript in order to produce usable and engaging interfaces. To that end, we will be exploring typography, multi-column layouts, navigation menus, image galleries, and more!

This course is designed to be complementary to *Core Studio: Interaction* (you should be enrolled in the studio with the same section as this course, section D, taught by Irwin Chen). The emphasis here will be placed on learning the technical things you'll need in order to explore the concepts you'll be learning in your studio class.

Class Website

I will be using a website to post homework assignments, class info and administrative-type stuff, resources, tutorials, notes, examples, etc. The url for the site is: http://interactionlab.info. You should check there throughout the semester.

Schedule

The following is a rough schedule of topics by week. It's likely that this will be adapted and adjusted as the semester progresses, so an up-to-date schedule will be available on the class website.

Week 1 (1/31)
HTML page anatomy and fundamental elements. Pointers for file organization and naming conventions. Working with relative paths. Uploading files to a remote server.

Week 2 (2/7)
Using CSS to control typography. Working with fonts on the web. Breaking a text document down into semantic HTML elements. Introduction to the *box model*.

Week 3 (2/14)
Grouping things together with generic *container* elements. Working with background images in CSS. Extending behavior using *:pseudo-classes*. Using *class attributes* for specificity. Introduction to *float* elements.

Week 4 (2/21)
Building multi-column layouts using *float* elements. The *clearfix* technique. Strategies for creating grid-based layouts.

Week 5 (2/28)
Visual effects and animation with CSS: rounded corners, text shadows, box shadows, transformations, and transitions.

Week 6 (3/7)
Layering elements on top of one another using relative and absolute positioning.

Week 7 (3/14)
HTML/CSS Quiz

Week 8 (3/21)
Introduction to Javascript and jQuery.

Week 9 (3/28)
SPRING BREAK.

Week 10 (4/4)

Handling mouse events using Javascript and jQuery. Building a tab-based navigation system.

Week 11 (4/11)
Working with HTML *form* elements. Handling key input events using Javascript and jQuery.

Week 12 (4/18)
Fluid layouts. Introduction to CSS *media queries.* Adapting layouts to different screen sizes and device orientations (for phones and tablets).

Week 13 (4/25)
Approaches/techniques for prototyping web *apps* for mobile devices.

Week 14 (5/2)
In-class workshop for final projects.

Week 15 (5/9)
In-class workshop for final projects.

Week 16 (5/16)
Parting remarks.

Student Expectations

You will be getting homework every week that is usually due by the following class. There may one or two larger assignments for which you will have more time (2–3 weeks).

There will be an in-class midterm quiz. Details about this will come in advance.

You are required to keep a website for this class, where you will create links to your completed homework assignments. We will discuss the details of this in the first week.

We will practice and work on things a good amount during class. You are welcome to bring a laptop to work on, but note that it will be fine to use the computers in the classroom. Details about the software tools we'll actually be using will be discussed in depth, but everything you'll need is completely free and available in computer labs on campus.

There is a lot of software out there, such as Adobe Dreamweaver, that enables one to produce websites through a visual interface, rather than by writing any code. In order to gain an understanding of how things really work under the hood, you will be required to write the code using a text editor.

The material we are covering in this course can be tough to learn. It's challenging to write code that doesn't have errors, and one small error can cause an entire project to break. It takes practice and patience to get into a groove with this stuff, but if you stick to it I think you'll find that you can make a lot of cool stuff with code that doesn't have to be extremely complicated.

Learning Outcomes

Core Lab is an introductory course and will not cover nearly all there is to know about developing web sites, but by the end of this course you will be able to produce simple, well-crafted user interfaces using basic hand-coded HTML, CSS, and Javascript. You will be comfortable analyzing design of a webpage layout and creating a prototype representation of it with code. You will have knowledge about some common CSS techniques, user interface patterns, and models for interaction design.

Grading

Your final grade will be based on the quality and execution of your homework assignments, the website that you keep for this class, your midterm quiz, how hard you work *in* class, and attendance.

Homework: 50%
Website: 10%
Quiz: 10%
In-class effort: 20%
Attendance: 10%

Attendance

Attendance is important. If you are absent more than *three* times, you will be at risk of failing this course.

Academic Integrity

Plagiarism and cheating is really bad.

The assignments given in this class will generally be exercises that involve writing HTML, CSS, and Javascript, and for these technologies it's easy to look up solutions on the internet, and copy somebody else's code wholesale. You won't learn anything by doing this.

That said, we may be leveraging open source frameworks and libraries to build things in this class. It's totally okay to incorporate example code that you find (or that I provide) in your work. But if you do this, have a basic understanding of why you are using a certain piece of code, make the effort to adapt it to the specific needs of your work, and always give credit where it's due.

Student Disability Services

In keeping with the University's policy of providing equal access for students with disabilities, any student with a disability who needs academic accommodations is welcome to meet with me privately. All conversations will be kept confidential. Students requesting any accommodations will also need to meet with Jason Luchs in the office of Student Disability Services, who will conduct an intake, and if appropriate, provide an academic accommodation notification letter to you to bring to me. At that point I will review the letter with you and discuss these accommodations in relation to this course. Mr. Luchs' office is located in 79 Fifth Avenue, 5th floor. His direct line is (212) 229-5626 x3135. You may also access more information through the University's web site at [http://www.newschool.edu/studentservices/disability].