Programming web-based educational media

Matthew X. Curinga

Antonios Saravanos

Table of Contents

**EDT 603 *Programming web-based educational media*, Spring 2019**

<!DOCTYPE html>  
<html>  
 <body>  
 <blockquote>  
 Anyone who has lost track of time when  
 using a computer knows the propensity  
 to dream, the urge to make dreams come  
 true and the tendency to miss  
 lunch.<br>  
 <strong>Tim Berners-Lee</strong>  
 <em>, inventor of the world wide web</em>  
 </blockquote>  
 </body>  
</html>

**Description:** In this course students learn techniques of web programming to develop interactive, educational media. Using the Javascript programming language and related web development technologies (HTML5, CSS, SVG), students gain practice in the programming and design of interactive software.

**Key words:** computer science, web development, mobile web, interaction design, html, html5, css, javascript, OOP, mobile first

# Office Hours

**Matt Curinga, Alumnae Hall, Room 226A**

* Monday, 11-1:00PM
* Tuesday, 2:30-4:30PM
* Thursday, 3-5PM
* *office hours by appointment*

# Goals and Objectives

This course builds on CSC 602 to move beyond basic programming concepts; students will gain expertise in building more complex computer programs, over several iterations. At the end of the course, students will be able to design educationally sound web-based learning media, solve moderately complex problems using OOP and functional programming paradigms, collaborate on team programming projects, and identify methods for *teaching* programming and web development.

Specific teaching and learning goals include:

* designing web-based interactions and multimedia to support learning
* coding effective user interfaces for learning
* implementing Universal Design goals for accessible web sites
* identifying effective methods for teaching more advanced programming concepts and web design skills

Specific software development goals include:

* modeling real world problems with software
* iterative software development
* testing and debugging
* Object oriented programming concepts:
  + Abstraction
  + Encapsulation
  + Objects & Classes
  + Composition
  + Inheritance
  + Polymorphism

# Course textbook

Curinga, M. Peter Wentworth, P., Elkner, J., Downey, A, and Meyers, C. (2018). [Think Javascript](https://mcuringa.github.io/think-js/). [free open textbook]

# Online Documentation

* [Mozilla Developer Network Javascript Docs](https://developer.mozilla.org/en-US/docs/Web/javascript)
* [Mozilla Developer Network HTML Docs](https://developer.mozilla.org/en-US/docs/Web/HTML)
* [Mozilla Developer Network CSS Docs](https://developer.mozilla.org/en-US/docs/Web/CSS)
* [React Framework](https://reactjs.org/docs/getting-started.html)
* [Bootstrap](https://getbootstrap.com/docs/4.2/getting-started/introduction/)

# Required Software/Online Accounts

* Software
  + [Slack](http://slack.com) (recommend desktop and mobile clients)
  + [Firefox web browser](https://www.mozilla.org/en-US/firefox/new/)
  + Chrome or [Chromium](https://www.chromium.org/Home) web browser
* Accounts
  + [Repl.it](https://repl.it)
  + [AU Ed Tech #code](https://auedtech.slack.com/signup)

# Recommended Books

*Not required, but a good book for the basics of HTML and CSS*

Duckett, J. T. (2011). [*Html & css: design and build websites*](http://htmlandcssbook.com/). Indianapolis, IN: Wiley Pubishing, Inc.



# Schedule

|  |  |  |
| --- | --- | --- |
| Module | Topic | Homework/Due |
| 0 | Preparing for the class |  |
| 1 | How the web works |  |
| 2 | Structure, data, style, logic | HTML mock-up of *Resume* |
| 3 | Mobile First, Styles, & Bootstrap | *Resume* |
| 4 | Usability, UL, forms | *Mini App* Pitch |
| 5 | Unit testing (no) |  |
| 6 | Files, objects, & storage |  |
| 7 | *Studio* |  |
| 8 | Mini App UX Testing | *Mini App* Due |
| 9 | 1:1 meetings |  |
| 10 | Node & Express | *App* Pitch |
| 11 | Routes & Navigation |  |
| 12 | Async calls & remote data |  |
| 13 | Searching, sorting, filtering |  |
| 14 | *Studio* |  |
| 15 | App Reviews | *Final App* & Presentation |

### Preparing for the class

It’s time to get ready for class. In this module you will take care of everything necessary to start the semester running and ensure you get the most out of your studies. There are a few things *everyone must do* before class starts, and I also list a few things you might want to review.

*Learning goals:*

* review basic skills for participating in an online class
* review prerequisite programming concepts
* learn to code static web pages with HTML and CSS

*To do:*

* [Join the AU Ed Tech Slack](https://auedtech.slack.com/signup)
* [Create an account on repl.it](https://repl.it)
* Complete the basic skills for online study:
  + record and upload a video (with decent lighting and audio)
  + record and upload an audio recording
  + take a screenshot (and post it)
  + take a screencast (and post it)
  + start a Google Hangout/Meeting (and be able to share your desktop, i.e. “present”)

*To review:*

* Examples and exercises in [*Think Javascript*](https://mcuringa.github.io/think-js/), especially arrays and objects/maps
* Self-study how to make basic web sites. I recommend [Intro to HTML/CSS: Making webpages](https://www.khanacademy.org/computing/computer-programming/html-css) on Khan Academy as a good place to start.

### How the web works

### Structure, data, style, logic

### Mobile First, Styles, & Bootstrap

### Usability, UL, forms

### Unit testing

### Files, objects, & storage

### *Studio*

### Mini App UX Testing

### 1:1 meetings

### Node & Express

### Routes & Navigation

### Async calls & remote data

### Searching, sorting, filtering

### *Studio*

### App Reviews

# Assignments & Grading

|  |  |
| --- | --- |
| Assignment | Points |
| Multimedia Resume | 30 |
| Mini App | 30 |
| Web Application | 30 |
| Self Evaluation | 10 |
| Group Evaluation | 0-3 extra credit |

# Books and online resources

### Documentation & Reference websites

* [Mozilla Developer Network](https://developer.mozilla.org/en-US/)
* [World Wide Web Consortium](http://w3.org)
* [W3 Schools](http://www.w3schools.com/)
* [Regular Expressions](http://www.regexr.com/)

### Books

* [HTML and CSS: Design and Build Websites](http://www.wiley.com/WileyCDA/WileyTitle/productCd-1118008189.html), our textbook
* [JavaScript & jQuery: Interactive Front-End Web Development Hardcover](http://www.wiley.com/WileyCDA/WileyTitle/productCd-1118871650.html), also J. Duckett, same series
* [Dive into HTML 5](http://diveintohtml5.info/) [free online]
* [The Elements of Typographic Style Applied to the Web](http://webtypography.net/toc/) [free online]
* [Mastering Regular Expressions](http://shop.oreilly.com/product/9780596528126.do)

### Tutorial websites & online learning

* [Code Academcy](http://www.codecademy.com/)
* [P2PU School of webcraft](https://p2pu.org/en/schools/school-of-webcraft/)
* [Treehouse](http://teamtreehouse.com/) [paid]
* [Thinkful](http://www.thinkful.com/)
* [GeekCamp::HTML5 Tutorial](http://www.geekchamp.com/html5-tutorials/1-html5-overview)
* [SkilledUp::Learn Web Design](http://www.skilledup.com/learn-web-design-guide/)

### Design, accessibility, UX

* [A List Apart](http://alistapart.com/topic/html)
* [Smashing Magazine](http://www.smashingmagazine.com/)
* [Adobe Kuler](https://color.adobe.com/create/color-wheel/)
* [Nielsen/Norman Group](http://www.nngroup.com/articles/)
* [United States Section 508](http://en.wikipedia.org/wiki/Section_508_Amendment_to_the_Rehabilitation_Act_of_1973)
  + <https://www.section508.gov/>
  + <http://webaim.org/standards/508/checklist>
* [Usability.gov](http://www.usability.gov/index.html)
* [Research-Based Web Design & Usability Guidelines](http://www.usability.gov/guidelines/guidelines_book.pdf)
* [hex/html color chart](http://www.december.com/html/spec/color.html)

### Online Tools

* [w3c HTML Validation Service](http://validator.w3.org/#validate_by_uri+with_options)
* [w3c CSS Validation Service](http://jigsaw.w3.org/css-validator/)

### Media Resources

* [Creative Commons Search](http://search.creativecommons.org/), for images, music, etc
* [Wikimedia Commons](http://commons.wikimedia.org/wiki/Main_Page), images and other media (including stuff from Wikipedia), curated
* [Open Clip Art](https://openclipart.org/), free vector graphics
* [Creative Commons Music](http://creativecommons.org/music-communities)
* [Fossil Bank](http://fossilbank.wikidot.com/)
* [Colour Lovers Palettes](http://www.colourlovers.com/)
* [Google Fonts](https://fonts.google.com/)

# Bibliography

Castro, Elizabeth. 2010. *Html 5 Visual Quickstart Guide.* Peachpit Press. ISBN 9780321719614.

Hayward, J. (2011). *Django JavaScript Integration: AJAX and jQuery*. Packet Publishing. ISBN 1849510342

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. M. (1994). *Design Patterns: Elements of Reusable Object-Oriented Software* (1st ed.). Reading Mass.: Addison-Wesley Professional. ISBN 0201633612.

Knuth, Donald. 1997. *The art of computer programming.* Addison-Wesley Pub. Co. Reading Mass. ISBN 9780201896831.

Moreno, R., & Mayer, R. (2007). Interactive multimodal learning environments. *Educational Psychology Review*, *19*(3), 309–326.

Moreno, R. (2006). Learning in High-Tech and Multimedia Environments. *Current Directions in Psychological Science*, *15*(2), 63 -67.

Shneiderman, B. (2000). Universal usability. *Communications of the ACM*, *43*(5), 84–91.

Shneiderman, B. (2002). Promoting universal usability with multi-layer interface design. *ACM SIGCAPH Computers and the Physically Handicapped*, (73-74), 8.

Teague, Jason. 2011. *CSS3: Visual QuickStart Guide.* Peachpit Press. Berkeley CA. ISBN 9780321719638.

Zelle, John. 2004. *Python programming: an introduction to computer science.*Franklin Beedle. Wilsonville, OR. ISBN 9781887902991.