GEORGETOWN UNIVERSITY
School *of* Continuing Studies
Journalism

MPJO/MPPR-779-40: WEB DEVELOPMENT FOR MEDIA

GEORGETOWN UNIVERSITY: MPS-JOURNALISM

Wednesdays, 5:15 p.m. to 8:15 p.m. | Summer 2017

Instructor: Greg Linch
Downtown campus, Mac lab, room C226

- Office hours are by appointment only

COURSE OVERVIEW

Merely using the web and digital tools is no longer enough for today's media professionals. Journalists and communicators alike need to have a strong foundational and practical understanding of how websites and web applications are built and how to troubleshoot when problems arise. This class does not aim to make you full-time coders (though maybe you'll want to be after this class!) or require any previous web development experience, but we do want you to come away with some coding skills. You'll also be able to collaborate more effectively with web developers and to continue learning on your own.

Students will learn about the various areas of web development and the fundamental technologies used to code and design web pages by diving into HTML and CSS, plus some JavaScript/jQuery and PHP. Students will set up and customize their own self-hosted website using WordPress. Readings, hands-on demonstrations, self-learning tutorials and assignments will be the basis for instruction.

COURSE OBJECTIVES

By the end of the semester, students should be able to:

- Identify the fundamental concepts and technologies underpinning modern web development, what role they play and how they work together
- Develop, deploy and maintain a dynamic website
- Use online tutorials and technical documentation to gain web development skills
- Synthesize technical knowledge to communicate ideas to non-technical people and non-technical ideas to technical people

REQUIRED READING

There is no required text for this course. Instead, the instructor will assign selected readings throughout the semester. Students will be expected to read materials before class, write an analysis post each week (unless otherwise noted) and discuss in class.

EQUIPMENT/SOFTWARE

Required:

Access to a computer (preferably a laptop) where you can install free software programs**:

- Sublime Text 3 (all platforms)
- GitHub desktop (Mac and Windows)
- MAMP for Mac or XAMPP for Windows

For the final project, you must also purchase (if you don't already have) later in the semester:

- A domain (e.g. your name if it's a portfolio site)
- Web hosting (e.g. basic tier with FTP access and the ability to manually install software)

Free accounts are required for:

- GitHub
- Codecademy
  - Note: You are not required to get a paid account or do the projects.

**Notes on required software:
- These three programs must be downloaded and installed *before* the first class to be sure you can hit the ground running.
- If you have problems installing that you can't resolve by searching online, please contact the instructor as soon as possible.
- After you install them, you don't have to start using them before class starts, but you're encouraged to explore what they do and why we'll be using them.
- **Important:** Don't change any MAMP or XAMPP settings unless instructed.

ATTENDANCE

As outlined by the university, missing more than two classes will result in a final grade reduction of one level (for example, an A will be converted to an A-). Absences for classes, beyond the initial two, will result in further reduction of the final grade. If you are absent for more than four classes, you will be in danger of failing this course.

If, for whatever reason, you are unable to attend class, please obtain notes and assignments from other students for the material you missed. Please show up to class on time. If you repeatedly show up tardy, it will have a negative impact on your grade — and you'll look bad in front of your fellow students.

CLASSROOM ETIQUETTE, CLASS PARTICIPATION AND OTHER GUIDELINES

All mobile devices must be silenced before class starts.

Students are expected to complete all the readings, self-directed learning and analysis posts before each class so they're ready to learn additional concepts and to apply skills to the hands-on work in class. These are all due before the next class to allow everyone to read each other's posts and to allow the instructor to best tailor the next lesson accordingly.

Whether you're using a lab computer or a personal laptop, computers in class will only be used for coursework. Your full focus and attention are required for this class. It is easy to miss important ideas or not fully grasp concepts if you are distracted. We're going to cover lots of technical topics during long class periods, so you don't want to miss anything.

Email is the primary means of communication outside of class for any questions. Phone calls should be arranged by email. If you have issues setting up the software we'll be using, try to troubleshoot on your own at first, but also don't wait until the last minute to contact the instructor in case you need assistance. For coding issues, figuring out why something doesn't work -- and how to fix it -- is a key concept of the class. You should exhaust all options for finding a solution (Google, StackOverflow, online tutorials and blog posts) before asking for help.

Class participation is an integral part of this class and accounts for 10 percent of your grade. That includes showing up and participating in class discussions, sharing your take on the weekly assignments and contributing thoughtful insights to the conversation. Engage in the class! You will get more out of it and so will the class.

Class discussions should be respectful and considerate of others' views and opinions. Everyone should feel comfortable asking questions and sharing their thoughts freely.

Instructional continuity: In the event of a weather emergency (or any other widespread emergency) that would close the Georgetown Downtown building, we will plan to meet virtually through online videoconferencing tools. More information will be provided on how this will work later in the semester.

ASSIGNMENTS

****** NOTE:** *Unless otherwise specified, all assignments are due* <u>*Sunday at 5 p.m.*</u> *before the next class.*

Late work will result in one full letter grade lower for each day after the deadline. If you have an emergency and will miss a deadline, you must request an extension from the instructor as soon as possible. If the request is approved, a new deadline will be given.

Self-learning: Read texts, watch videos or complete tutorials as assigned. *Due: Sundays by 9 p.m.*

Analysis/update posts (~300-400 words, 11 total): For assigned readings and self-learning, write reactions and highlight what you found most interesting or had questions about on the course blog. These posts should be a mix of synthesis to show your understanding of the material as well as like a journal of your progress (here's what I did, here's what I'm going to do, here's what hurdles exist, etc.). These are generally due a day after the assignment is due so you have time to reflect on what you learned. *Due: Mondays by 10 p.m., unless otherwise specified*

Mid-term: Project explainer or developer profile (~800 words): Pick a project related to your course of study to deconstruct what, how and why it was built, plus lessons learned or what it might mean in the larger context of the field. Alternatively, you may profile a developer working in news, PR or – perhaps more interestingly – outside of news and PR whose work might be relevant to those fields. The project should not be something you or your company worked on and the developer should not be a friend. Be sure they know this will be published on the class site. *Due: Sunday, June 25 at 5 p.m.*

Development environments: Having local and production development environments are essential to your final project. You need to set up a local environment for testing on your computer and a production environment on a hosting service for the public-facing site. The local environment will use MAMP for Mac or XAMPP for Windows. The production version will use a hosting account and domain you've purchased. For the hosting, you must be able to install WordPress manually, use FTP and create a MySQL database. *Due: Sunday, July 9 at 5 p.m.*

Final project site: Create a customized project using the WordPress site created for the production environment. This will be preceded by a project pitch detailing your plans, including what customizations and features you'll be adding. For example, you might do some combination of creating a child theme, building a new plugin, using an API, etc. If your pitch is late, it will affect your final project grade. *Pitch due: July 2 at 9 p.m. Project due: Sunday, July 30 at 9 p.m.*

Final project critiques: Evaluate your peers' project sites with constructive feedback. This are due soon after the demos so everyone has a chance to incorporate feedback over the weekend. *Due: Thursday, Aug. 3 at 9 p.m*.

Concept/skill/tool explainer (500-600 words): What did you find most difficult to learn this semester? Take that concept, skill or tool and write an explainer post you could share with a friend or coworker to break it down for them as a non-developer. *Due: Sunday, Aug. 13 at 9 p.m.*

Surveys: Fill out this survey before the first session (http://bit.ly/MPJO779SUM17survey1) and any others sent out during the course based on specified deadlines. These surveys will help the instructor know where everyone stands and if any adjustments in content or pacing are needed.

GRADING

Your course grade will be based on the following:

In-class participation/surveys:	10%
Self-directed learning/Codecademy:	10%
Analysis/update posts:	10%
Development environments:	5%
Midterm - Project explainer or developer profile:	15%
Final project site:	30%
Final project critiques:	10%
Concept/skill/tool explainer:	10%
Total:	**100%**

Graduate course grades include A, A-, B+, B, B-, C and F. **There are no grades of C+, C- or D.**

A	100-93	B-	82.99-80
A-	92.99-90	C	79.99-70
B+	89.99-88	F	69.99-0
B	87.99-83		

The instructor will provide a warning by mid-semester to any student who appears to be on track for a poor final grade.

UNIVERSITY RESOURCES

Georgetown offers a variety of support systems for students that can be accessed on main campus or at the downtown location:

- MPS Writing Resource Program
  202-687-4246
  http://writingcenter.georgetown.edu/

- Academic Resource Center
  202-687-8354 | arc@georgetown.edu
  http://ldss.georgetown.edu/

- Counseling and Psychiatric Services
  202-687-6985
  http://caps.georgetown.edu/

STUDENTS WITH DISABILITIES POLICY

Students with documented disabilities have the right to specific accommodations that do not fundamentally alter the nature of the course. Students with disabilities should contact the Academic Resource Center (Leavey Center, Suite 335; 202-687-8354; arc@georgetown.edu; http://ldss.georgetown.edu/index.cfm) before the start of classes to allow time to review the documentation and make recommendations for appropriate accommodations. If accommodations are recommended, you will be given a letter from ARC to share with your professors. You are personally responsible for completing this process officially and in a timely manner. Neither accommodations nor exceptions to policies can be permitted to students who have not completed this process in advance.

GEORGETOWN HONOR SYSTEM

All students are expected to maintain the highest standards of academic and personal integrity in pursuit of their education at Georgetown. Academic dishonesty in any form is a serious offense, and students found in violation are subject to academic penalties that include, but are not limited to, failure of the course, termination from the program, and revocation of degrees already conferred. All students are held to the Honor Code. The Honor Code pledge follows:

In the pursuit of the high ideals and rigorous standards of academic life, I commit myself to respect and uphold the Georgetown University Honor System: To be honest in any academic endeavor, and To conduct myself honorably, as a responsible member of the Georgetown *community, as we live and work together.*

PLAGIARISM

Stealing someone else’s work is a terminal offense in journalism, and it will wreck your career in academia, too. Students are expected to work with integrity and honesty in all their assignments. The Georgetown University Honor System defines plagiarism as "the act of passing off as one's own the ideas or writings of another.” More guidance is available through the Gervase Programs at https://honorcouncil.georgetown.edu/system/policies. If you have any doubts about plagiarism, paraphrasing and the need to credit, check out http://www.plagiarism.org.

SYLLABUS MODIFICATION

The syllabus may change to accommodate discussion of emerging topics. Also, the schedules of guest speakers may require some shifting of our agenda. The instructor will make every effort to provide as much advance notice as possible for any alterations.

CLASS SCHEDULE

Pre-course preparation

- **Required:** *** Complete this survey ***
- **Required:** Reading/watch
  - How the Internet works
  - Timeline of software languages
  - Timeline of networking and the web
  - Computational thinking and journalism: part 1 and part 2
  - WordPress software philosophy
  - GNU free software philosophy
  - Unix philosophy
  - The Zen of Python
  - The Pragmatic Programmer Quick Reference Guide
- **Required:**
  - Download and install Sublime Text (all platforms)
  - Create a free GitHub account
  - Download and install GitHub for Mac or GitHub for Windows and connect it to your GitHub account

- **Required**: Write analysis post that includes why these ideas matter, what you found most interesting and your main takeaways from topics discussed
  - Include a link to your GitHub profile in the post

Week 1 - May 24
Overview of class and how the web works

- Overview of the class
- Review of survey results
- Discussion of pre-course reading materials
- How does the Internet work? How does the web work? What's the difference?
  - Internet protocols
  - Client-server model
  - What are front-end, back-end, full-stack?
- Overview of tools we'll use

Assignments for week 1

- Think about: What kind of development do you want to focus on? Track selection due with pitch: *July 2 by 9 p.m.*
- Think about: What do you want to build? Final project *due July 30*
  - What problem do you want to solve?
  - Focus on the *what*, not the *how* yet
- Read: Getting started with HTML and CSS
- Read: Basic intro to HTML/CSS for journalism students
- Create: Codecademy account
- Complete: Codecademy Learn HTML & CSS course lessons 1-2 on HTML
- Read: Meet Your Web Inspector
- Read: How to inspect styles
- Read: Understanding the GitHub flow
- Complete: Intro to GitHub exercise
- Analysis/update post
  - Include link to the test repo you made in the exercise
  - Include link to Codecademy profile (example)

Week 2 - May 31
Browser and front-end basics

- View source/Inspect element
- Document-Object Model (DOM)
- HTML revisited

- CSS overview
- Responsive web design discussion
- Git and GitHub discussion

Assignments for week 2

- Complete tutorial: Codecademy Learn HTML & CSS course lessons 3-8 on CSS
- Read: A List Apart - Responsive web design (focus on fundamentals, key concepts, what this entails)
- Read: W3 Schools - Responsive web design from “Intro” to “Video”
- Read: Getting your project on GitHub
- Create a prototype homepage (e.g. for a personal site) using HTML and CSS
- Add your prototype homepage to GitHub using the desktop app
- Analysis/update post

Week 3 - June 7

Scripting and databases basics

- The DOM revisited
- Client-side scripting vs server-side scripting
- Intro to programming concepts with JavaScript

Assignments for week 3

- Complete: Codecademy Learn JavaScript course lessons 1-5
- Finish JavaScript exercise from class
    - Use CSS, add styles
    - Using JavaScript/jQuery, add captions that show/hide for each photo
- Analysis/update post

Week 4 - June 14

Scripting, databases, servers, different code environments

- Review JavaScript/jQuery concepts and gallery exercise
- Intro to server-side scripting
- Intro to databases for web applications
- Intro to web servers and server stack
- Explanation of development, staging and production environments
    - MAMP (Mac) and XAMPP (Windows)
    - PHPmyadmin
- Intro to WordPress
    - Difference between dot com and dot org

- How to set up self-hosted WordPress on your local machine

Assignments for week

- Complete: Codecademy jQuery course lessons 1-4
- Read: GitHub features
- Analysis/update post

Week 5 - June 21
Project management

- What is agile? What is scrum? What is waterfall?
- What is user-centered design?
- How to manage technical projects
- Steps/checklist
- Using GitHub for projects
  - Issues
  - Pull requests
  - readme/wikis
- JavaScript/jQuery revisited
  - Start writing a gallery to add to the homepage from Week 2

Assignments for week 5

- Read: Agile manifesto
- Watch: Agile best practices
- Read: Shh! Don’t Tell Them There’s No Magic In Design Thinking
- Complete: Mid-term: Developer profile or technical project explainer
- Analysis/update post

Week 6 - June 28
Diving into development, part 1

- Loops and arrays
- More on version control with GitHub
- Intro to PHP
  - Adding PHP to the homepage you updated last week
- Intro to WordPress themes and plugins

Assignments for week 6

- Complete: Codecademy PHP courses 1, 2, 4, 5 (skip 3)

- Read WordPress documentation entry excerpts to better understand customization options. Focus on fundamentals, key concepts and general steps.
    - Theme development: read intro, “What is a theme?”, “Setting up a development environment”, “Template files” and stop at “Using template files”)
    - Child themes: read until “How to Create” and then briefly skim the rest
    - Writing a plugin: intro and “What is a plugin?”
    - Post formats: intro and “supported formats”
    - Post types: read through “custom post types”
    - Custom fields: intro
- Pitch post: 200 words on what you want to do for your final project site, what track you’re doing (front-end, back-end, full-stack), what customizations you’ll include, why you’re doing this and who the audience is. *Due July 2 by 9 p.m.*
- Bring your laptop to class on July 5 for local development environment configuration
- Analysis/update post

Week 7 - July 5
Diving into development, part 2

- Local development environment setup
- Codex/documentation review
- Diving deeper into themes, child themes and plugins
- Post formats, custom post types, custom fields — oh, my!

Assignments for week 7

- Complete: Finish setting up local dev environment (due by July 9 at 5 p.m.)
- Complete: Learn the Command Line course lessons 1 and 2
- Find and read code and documentation on GitHub for WordPress themes (example)
- Update pitch post, including some ways you could use the different available options for your project
- Work on project
- Analysis/update post

Week 8 - July 12
CLI: Command-line interface

- Command line basics discussion
- Discuss projects and what customizations everyone is planning
- Show example of how to create a custom post type
- Intro to APIs

Assignments for week 8

- Watch: What is an API?
- Read: What is an API? In English, please.
- Watch: REST API concepts and examples
- Read: WordPress REST API intro page
- Complete: Learn how to use the YouTube API "How to Use APIs" lesson
- Complete: Install WordPress on your hosting
- Work on your final project
- Analysis/update post

Week 9 - July 19

APIs: Application Programming Interfaces

- APIs discussion
- Project questions? Ask them!

Assignments for week 9

- Work on final project
- Analysis/update post

Week 10 - July 26

Final project workshop

- Work on your final projects
- Ask the instructor questions
- Ask your peers questions

Assignments for week 10

- Work on final project: *Due Sunday, July 30 at 9 p.m.*
- Analysis/update post

Week 11 - Aug. 2

Final project demos

- Share projects
- Take notes for feedback to provide as GitHub issues
- Discussion

Assignments for week 11

- Write a critique with constructive feedback for each site and send to instructor by *Thursday, Aug. 3 at 9 p.m*. The instructor will then compile the critiques, remove names and send by Friday night.
- Make adjustments to your project based on feedback over the weekend. Send update message to instructor when changes are finished by *Sunday, Aug. 6 at 5 p.m.*

Week 12 - Aug. 9
The wider world of web development

- McClatchy newsroom visit
- An overview of other web dev tools and what they are best used for

Assignments for week 12

- Complete: Self-learning plan
  - You've heard more about what more there is to learn, besides diving deeper into what we've started using in class. Now it's time to create a plan for advancing your skills after the class. Devise a self-learning plan to use after the course ends, including goals and timetable, then post it to the class blog in lieu of an analysis post
- Complete: Project explainer or professional dev profile: *Due Sunday, Aug. 13 by 9 p.m.*
- Complete: End-of-course survey

Week 13 - Aug. 16
Leveling up your skills

- Review self-learning plans and next steps
- Feedback and discussion of goals
- Discuss end-of-course survey

Assignment for week 13

- Analysis/update post: Big-picture class thoughts and feedback