ART S167:
Intro to Interactive Design: *Ease into Digitality*

Yale Summer Session
7/1–8/2/2024 (Session B)

Instructor:
Mianwei Wang
mianwei.wang@yale.edu

Overview

This summer studio course is about thinking and making work in the web browser. We will use the web and the internet as an entry point to reflect on our digital experience, approach websites as our individual practice, and retrospect on the development of our digital life so far. Combining readings, writing prompts, workshops, and discussions, we will progress by completing each assignment and project, exploring the nature and potential of webpages for our use.

This 5-week course will cover foundational front-end languages HTML, CSS and JavaScript. Critiques and discussions will focus on the choices in content, layouts and other forms of the design, learning path is provided through in-class workshops and exercises, with out-of-class assignments and office hour meetings. No prior programming experience is required.

Course Objectives

- We learn how to learn: We explore and study how we learn individually and collectively, and figure out what works the best for each of us.
- Technical: Master the techniques and tools in creating and publishing a website project. Gain fluency in HTML, CSS, and JavaScript, and develop strategies for troubleshooting.
- Technical: Apprehend a general approach to programming. By the end of the course, you should be able to determine the best way to structure and begin a project; know how to write efficient, eloquent code;
- Literacy: Be critical and inventive when designing for the web.
- Creative: Realize projects that make compelling use of content, visual design, and technical opportunities.
- Creative: Develop personal design strategies and interests that can be applied to a studio practice more broadly.

Tools

- Computer: a Mac* or PC is required for in-class exercises
- Web browser: Google Chrome*, Firefox
- Text editor: Visual Studio Code* / Sublime text / Glitch…
- Other softwares and services: GitHub.com, netlify.com, are.na, Dropbox and Google sheets etc

* used in class demos

Projects

Walk								
		Channel						
							Easter Egg	

P1: Walk

Create a website that journals a walk of yours. The site should feature multiple locations, scenes, paths, and/or conversations with people you met during the walk. Collect metadata from your walk, including:

- The starting and ending times
- Geolocations
- Objects
- Sounds
- Feelings
- Timestamps of the feelings

Take note of subtle activities in the space, such as room tone, bird chirping, breeze, or view changes over time. If you take a video of the walk, consider how the video would introduce your personal experience to the audience. Similarly, think about how text and other formats can convey the experience.

What opportunities do you have in creating a walk on web? And how is it different from the web experience to the initial walk? Utilize data from your first-person perspective, including your internal and external activities, and/or compile metadata about the environment, such as time and spatial descriptions.

We will begin by creating an HTML-only version of the site, then adding CSS. Given the timeframe of this project, text-based methods are recommended, but you are welcome to incorporate all formats including images, audio, video, or any combination thereof.

P2: Channel

A website is a container for much of the content we choose. The content can define the container, and vice versa. In this exercise, we will start by creating a collection of content first, and discuss what the space could be like for the content to live.

We will look at the website as a publishing tool and create a Content Management System (CMS) based website. During this project, we will investigate our content in depth, look for internal relatedness, and design an environment that situates such content.

Curate an Are.na channel and use the Are.na API to deploy it to a website. We will spend the first few meetings on the content of the channel, then explore the design in CSS. After the introduction to JavaScript basics, we will utilize ChatGPT-powered tools in the learning path. Aside from debugging and technical work, our discussion will center around how your design reflects your content curation. Consider:

- what are the attributes that work in our favor within the constraints of public channels?
- who can view the channel?
- who can contribute to the channel?
- where are you collecting the content from, and
- how long can the content, and the channel live?
- will the channel evolve over time? Will the website do so as well?
- how does the choice of using Are.na affect your answers to the questions above?
- and would choosing a different Content Management System change your answers?

P3: Easter Egg

In this project, we will look at our everyday surroundings and reimagine the connections and interactions within our space. Add an Easter egg to your channel website (P2) and create a physical component as a spatial intervention on campus (Green Hall).

On the website, the Easter egg can be triggered by physical parameters such as time, location, temperature, or interactions on the website, like media queries or certain inputs.

For the physical component, consider where and how the Easter egg can be installed. Look for a specific space that suits you best in terms of its physical structure, such as corners or staircases. Also consider factors like lighting, traffic, and various conditions. We will conduct a "scavenger hunt" for the Easter eggs in the final class, and the installations can be projected, accessed via a QR code, displayed on a monitor, and so forth.

Attendance

Attendance at all class meetings is mandatory, including on all holidays. Three or more unexcused absences will result in a failing grade for the course. Three or more late arrivals (more than 15 minutes late) equals an absence. If you absolutely must miss class, please email me in advance.

In addition to your attendance record, any work or class performance (discussions or presentations) missed during your absence will also count toward your final grade.

Academic Integrity

Students will become familiar with using pre-existing language, images, and software as raw material while creating entirely new works. While making websites, we will learn which technologies could be appropriated and how to properly credit their inclusion.

From academic integrity at MIT: "Writing Code" (https://integrity.mit.edu/handbook/writing-code):

> "Writing code is similar to academic writing in that when you use or adapt code developed by someone else as part of your project, you must cite your source. However, instead of quoting or paraphrasing a source, you include an inline comment in the code. These comments not only ensure you are giving proper credit, but help with code understanding and debugging. ... You should not simply re-use code as the solution to an assignment. Like academic writing, your code can incorporate the ideas of others but should reflect your original approach to the problem."

Try retyping someone else's code instead of copy and pasting it. Be careful about pasting huge blocks of code. remember to do things one step at a time so you truly understand each piece of code's unique function.

This document is tentative and subject to change according to actual course operations. Additional information and materials are available through email. Last updated on Mar 4, 2024.