Generative Art Online

MADD 23632

Instructor:	Nick Briz (nbriz@uchicago.edu)
AI-TA:	netnet
When:	Tue : 03:30 PM-06:20 PM
Where:	Logan Center 014
Office Hours	Edelstone (6030 S Ellis Ave) #267 by appointment

Course Description

Trade doomscrolling for art making. The web wasn't built for mindless consumption and targeted ads, it began as an open, accessible space for human connection and creative expression. In this course, you'll help reclaim that original spirit. You'll learn the fundamentals of creative coding and apply them to one of today's most malleable digital canvases: the web browser. Through the process of creating generative and interactive Internet art, you'll develop a foundational understanding of core programming concepts as expressed through JavaScript, the web's de facto programming language. We'll learn how to produce work that responds to various input sources (trackpad/mouse, touchscreen, keyboard, cameras, microphones) and how to fetch and incorporate data from external APIs elsewhere on the Internet. In this course we will not only push the boundaries of what the web can *do* but also challenge what we think the web is *for*.

Learning Goals

- Foundational understanding of web programming concepts, tools and working knowledge of the JavaScript programming language.
- General understanding of the history and theories behind generative and interactive art as well as core principles of its practice.

Class Materials

In order to participate in this course you will need to have a decent computer (desktop or laptop with 8-16GB of ram or more) and a modern Web browser like Firefox, Brave, Chrome or others (**do not use** Internet Explorer or Safari, those are subpar browsers). Personally, I will be demoing things in class using Firefox.

You will also need to create a free GitHub account, this is where you'll be uploading your projects (the actual code) before submitting them on Canvas. (*if you are new to GitHub, consider signing up for the GitHub Student Developer Pack*)

You will also need a code editor. For the majority of the quarter I will be demoing concepts and creating examples/sketches using netnet.studio. This is the tool we will use for all of our initial "sketches" and can also be used to create the final project. However if you have familiarity with a modern code editor like Pulsar, Sublime, or VSCode you can use any of those for the final if you prefer (so long as you know how to get your work published on GitHub using those editors)

Class Discussions

So much of what we're going to cover in class, both in terms of the theory and practice, can be gleaned through your own online research. The most valuable aspect of learning this material in the classroom, rather than on your own, is the chance for real-time interactivity with your professor and peers. I can not stress enough how important it is to take advantage of class discussions. At times these will be technical discussions about how the Internet and the Web work, coding tricks and techniques, and any other topic relating to our craft. They will be theoretical discussions about concepts and ideas explored through the many case studies of creative works we'll be looking at throughout the course.. And they will also be "critiques", where we'll look at each other's creative sketches, discuss our creative decisions, and give each other feedback.

If you need to miss class for a legitimate reason (medical or family emergency, professional opportunity, etc), send me an email ahead of time so that I can mark it as an excused absence. Attendance and participation in class is **20% of your final grade.**

Assignments + Evaluations

4 Creative Sketches

Each student will be expected to complete and submit 4 online generative art “sketches” on the dates specified below. These assignments are creative coding experiments, first drafts with potential for further iteration. These will be created on netnet.studio and saved as “sketches”, the URLs for which should be submitted to the class canvas.

These projects will be responses to “case studies”, works of generative art we’ll discuss in class and use as jumping off points for our own creative experiments. Each week (during weeks 2 - 5) I will be introducing new techniques and concepts, often building on the previous weeks, and starting on these code sketches together in class.

In my evaluations I will be reviewing both craft (how you’ve written your code) and concept (the idea behind the sketch), I will either mark the assignment as “complete” or “incomplete” on canvas, if/when I mark an assignment as “incomplete” I will leave feedback with clear request for changes you will need to make. Once you’ve made those changes you can resubmit the sketch and I’ll then either mark the assignment as “complete” or leave more feedback.

These “sketches” are worth 60% (15% each) of your final grade and must be “complete” in order to receive full credit.

1 Final Project

The final project will be a finished, exhibition ready, work of online generative art. It should extend the work you've created earlier in the quarter, by either combining and/or iterating on one (or more) of your prior assignments. Unlike your sketches (which are open ended explorations of generative art concepts and techniques) your final project needs to have a stated goal, context and title (Details will be discussed in class). **The final is worth 20% of your grade**, to receive full credit the work will be submitted on canvas as a published URL and must also be presented in class on either week 8 or 9 for an in-person ~15 min critique. *NOTE*: if a final requires alternative hosting options you are welcome to host your project elsewhere (so long as it has a publicly accessible URL), in these instances you should submit both the public URL as well as the URL to your GitHub repo (so I can review the code).

Course Schedule

week 1: 03.24.2026

Introduction to Generative Art

We'll also discuss the history (and pre-histories) of Generative Art, its core ideas and how we'll be engaging with these concepts in this course. We'll introduce the online coding platform that we'll use to create our work this quarter as well as how/why we'll be engaging with the web specifically as a creative canvas.

week 2: 03.31.2026

Case Study: 10print

We'll discuss early generative artists (1960s) like Vera Molnár and the role "randomness" played in their work. Through a close reading of one of the most iconic works of generative art, 10print (1982), we'll introduce the core concepts of programming as well as a couple of the principle techniques for creating generative systems, namely randomness and recursive animation loops.

week 3: 04.07.2026

Case Study: Emoji.Ink

We'll discuss the genre of software art or "artware" and take a close look at the piece emoji.ink (2015) by tig.ht (Yung Jake and Vince Mckelvie). We'll learn how to make interactive works by introducing the paradigm of event based programming. We'll also discuss the concept of software libraries.

week 4: 04.14.2026

Case Study: To Make A Dadaist Poem

We'll discuss appropriation techniques from the dada movement, in particular the piece To Make A Dadaist Poem (1920) by Tristan Tzara. We'll explore "data driven" online algorithmic works and learn how to incorporate data pulled from third party APIs into our own pieces.

week 5: 04.21.2026

Case Study: A Tribute to Heather

We'll continue to build on the generative art techniques introduced in the weeks prior to create algorithmically generated (2D) visual compositions through a close reading of A Tribute to Heather (2013) by Evan Roth.

week 6: 04.28.2026

Case Study: *I Miss Every Sound I've Ever Heard, Does Anyone Feel The Same*?

We'll explore how the web browser can become a performative instrument and take a look at pieces like I Miss Every Sound I've Ever Heard, Does Anyone Feel The Same? (2022) by Chia Amisola and learn how they create systems for performing audio/visual compositions which break out of the browser window to incorporate pop-ups as formal elements.

week 7: 05.05.2026

final project: studio session
This week we'll have an "open studio day", after discussing the goals and expectations of the final project we'll use the rest of class time to discuss and start working on our finals.

week 8: 05.12.2026
final critiques (session I)
The first half of class will present their final projects for critique

week 9: 05.19.2026
final critiques (session II)
The second half of class will present their final projects for critique

AI Policy

We've entered the era of "machine learning" and generative AI (not to be confused with generative art). These algorithms are having (and will continue to have) drastic effects on every aspect of our society (including art). Today, artificial neural networks trained on troves of data (which are not always ethically sourced) can make "predictions" and create "hallucinations" that would have seemed like impossible sorcery just a few short years ago. In certain high stakes applications this can save lives, but it can also destroy them. In other contexts this biased hallucinatory predictive sorcery can be quite exciting, as is the case with media art. This technology, like many others that came before it (smart phones, the Internet, the computer) will most certainly change everything in our field, exactly how and to what extent is still anyone's guess. While our primary focus in this course will be to learn how to create generative and interactive online works by writing hand-crafted code, along the way we will also be discussing and experimenting with AI systems so that we can collectively learn how to (if we choose to) incorporate them into our practice in a way that augments both our learning and making, rather than undermine it.

Incorporating AI models in our work
A couple of different times throughout the quarter we will discuss how to incorporate AI models as one (of many) components in our generative art works. Details about how/why will be discussed in class when those tools/topics come up.

Leveraging AI models to create our work
In this class the most likely use of generative AI will be code generation/evaluation using LLM (large language models). You are encouraged to use these in ways the supplement your learning, rather than impede it. To ensure that is the case I ask that you avoid using any LLMs at the start of class and instead slowly incorporate them into your practice as we introduce specific and deliberate use cases throughout the quarter.