

Typography & Interaction

Fall 2026

The New School, Parsons, MPS CD
PMCD 5001, CRN 4253/9023
2 W 13th St., Room 1201
Thursdays, 4–6:40pm

Michael Fehrenbach

mfehrenbach@newschool.edu

Rijk van Zanten

rijk@newschool.edu

Course Description

Typography & Interaction is a year-long course, divided into two classes, which will provide a rigorous foundation of typographic and interaction principles in the context of digital design. Over both classes, students will acquire and hone the skills they need for success in the field of interactive design.

This first semester will focus on a mastery of type and layout concepts on the web:

Typography is the infrastructure of communication in nearly any visual medium. It provides the very first shape and form to written content, and as designers, it is our responsibility to practice this with intention and care. Whether towards goals of expression itself or in the service of ideas, the designer must understand type to use it successfully. In this way, we are stewards of meaning.

Digital design, the web in particular, is inextricably linked with typography—from the very letters of code at its base to the words in arrangement we see on a screen. Type, thus, is the scaffolding in which all interaction design first rises. The very shape of the web, in its layouts, systems, and patterns—and its various technologies—all exist in the service of type, at their root. They provide the tools with which we can breathe a form and different, digital life into that meaning.

In this class, students will learn intermediate and advanced methods in typography and layout as they concern interactive design. We will use web technologies as the lens to examine this subject—introducing the foundational, front-end languages of HTML (*HyperText Markup Language*) and CSS (*Cascading Style Sheets*) to achieve our designs. Students will understand the specific challenges of type in this medium, but also how it offers unique and particular forms to us as designers. They will learn the common tools and paradigms with which we practice, while developing their own visual, design vocabulary and critical understanding.

[Course Catalog](#)

Learning Outcomes

By the end of this semester, students will:

- Demonstrate advanced knowledge of and be able to critically analyze type, form, and interactivity as it applies to screen-based media.
- Understand how to effectively deploy type hierarchy in layout and grid systems, in responsive, device-agnostic design.
- Effectively translate these designs into functional websites using HTML, CSS, and other web technologies.
- Design and prototype work while taking into account the ever-shifting, bespoke challenges of web design.
- Give, receive, and respond productively to feedback and critiques both asynchronously and in-person.
- Think critically and develop their own, distinct thoughts on the role of digital within the larger canon of design.

Course Outline

The course is structured into thematic units, each bookended by readings on the subject and a project that will demonstrate the material:

Unit 1: *Type and the Web*

Weeks 1-6

We will focus on reviewing the core principles of typography, and introduce the web and its base technologies. Students will learn about HTML, semantic DOM, basic CSS, as well as type hierarchy and the use of typefaces on the web.

Readings

The Principles of the New Typography

Jan Tschichold, 1928

The Crystal Goblet, or Printing Should Be Invisible

Beatrice Warde, 1932

Detail in Typography

Jost Hochuli, 1987

The Elements of Typographic Style

Robert Bringhurst, 1992

A Handmade Web

J.R. Carpenter, 2015

Project 1: *Manuscript*

The unit ends with Project 1, *Manuscript*, which students will present for critique on October 1.

Students will choose a seminal design text from [readings.design](#), read and respond to it, and typeset their selection and reply together as a web page. Other texts are also allowed on a case-by-case basis.

We'll be looking at the quality of the responses, appropriate type selection and hierarchy, semantic HTML, and basic CSS.

Unit 2: *There Is No Perfect Layout*

Weeks 7-10

Students will learn how to design and implement more complex, flexible layouts, while collaborating closely with a classmate. We'll introduce responsive design, media query CSS, and advanced web type techniques.

Readings

Investigations on Gestalt Principles

Max Wertheimer, 1923

Continuity and Change

Max Bill, 1953

Grid Systems in Graphic Design

Josef Müller-Brockmann, 1981

The Web's Grain

Frank Chimero, 2015

The Diminishing Marginal Value of Aesthetics

Toby Shorin, 2017

Project 2: *Spread*

This unit concludes with Project 2, *Spread*, which students will present together on October 29.

Students will work in pairs, with the texts they selected in *Manuscript*. Each duo will sketch collaboratively and then implement a new expression together, via pair programming. The final web page will be responsive for mobile, desktop, and print layouts.

Here we're looking for successful design and development collaboration, box-model layout design, and the use of responsive media queries.

Unit 3: *Typography as Interface*

Weeks 11–15

In our final Fall unit, we will focus on creating advanced, multi-page layouts with grid systems, prototyping their flows, and exploring typography's usage as interface elements for navigating a website.

Readings

Design Interface: How Man and Machine Communicate

Gianni Barbacetto, 1987

A Software Design Manifesto

Mitchell Kapor, 1990

Typeface As Programme

Jürg Lehni, 2011

Interface Writing: Code for Humans

Nicole Fenton, 2014

*My website is a shifting house next to
a river of knowledge. What could yours be?*

Laurel Schwulst, 2018

Project 3: *Binding*

This unit, and the first semester, will culminate with Project 3, *Binding*, which will be presented in class on December 3.

Students will assemble a collection of texts from *Spread*, combined with their original selection, into a “book.” The book will take the form of a multi-page website with a homepage (cover), navigation (table of contents), individual pages for each text, and an introduction (colophon)—with a consistent design system applied across all pages.

Here we want to see effective multi-page design and site navigation, advanced layouts (flexbox, grid), consistency and expression across the pages and content, and overall polish/nuance.

Evaluation Criteria

Reading Responses

Each unit begins with a set of readings to introduce the subject. Students are expected to read the required selections and synthesize their own thoughts in a considered, written response prior to the next class. We are not looking for summarization, here—these should be personal reflections on the subjects, and are evaluated with this lens. We then discuss these readings as a group.

Exercises and Milestones

Each unit will also have specific, technical exercises and milestones that are assigned towards completion of the projects. These assignments are expected to be completed outside of class, before the next session. Some of these will be small; some of these will be large. They are *all* evaluated for completion and quality.

Engagement

Students are expected to actively and passionately participate in this course. This means more than showing up and turning things in on time—which should be a given. Beyond that baseline students should be curious, prepared, thoughtful, vocal, and intentional throughout the course. They should make us understand why they are here, and demonstrate to us that they care about themselves, their work, and each other—and ultimately, about this chosen profession.

This engagement will be unavoidably reflected in the quality of students' work—but we also evaluate this discretely based on their participation in and out of the classroom, with us and with their peers.

Projects

The bulk of the work for this class takes the form of projects. They are intended as opportunities for students to demonstrate the knowledge and skills learned in class while developing their own practice, and are evaluated in this light.

There will be check-ins and reviews around each of these before the final due dates, when we will have critiques as a group. In addition to the quality of the finished project itself, students will have in-person design and technical reviews—where they will be assessed on their explanation and understanding of their work, to determine whether they have met our learning objectives.

More specific, per-project evaluation criteria will be delineated with each project's introduction.

Grade Calculation

Reading Responses	10%
Exercises and Milestones	10%
Engagement	20%
Project 1: <i>Manuscript</i>	10%
Project 2: <i>Spread</i>	20%
Project 3: <i>Binding</i>	30%

Materials and Supplies

In the open tradition of the early web, the only materials required are a computer, a browser, a text editor, and an internet connection. The specifics of these are open to the student's individual preferences and practices. We will do our best to accommodate everyone and will make recommendations, when needed.

In class, we will demonstrate using Figma for visual design and sketching, Visual Studio Code for programming, and GitHub/GitHub Desktop for version control and our project hosting. All of these products are available for free, or offer free education licenses with New School emails.

We will use the following tools to organize our class:

Course Site

For housekeeping, agendas, and lectures.

Submission Form

For submitting your work/URLs—this is “turning it in.”

Slack Channel

For direct and asynchronous communication (*not* email).

Zoom Room

For screen sharing and recording.

Google Drive

For document collaboration and recorded lectures.

GitHub Organization

For code examples and sharing.

Figma Team

For visual sketching and collaboration.

Class Policies

Our Community

Our class will create and maintain an agreement, intended to help us foster a safe, empathetic, and productive space for our course. It is built on trust and accumulated experience across cohorts. It can be revised and modified, with all of our input, over the year, and lives on [our course site](#).

Inclusion

Our intent is to respect and give forum to a range of perspectives and backgrounds, including culture, race, gender, sexual orientation, socioeconomic status, disability, and age. In instances where we are not personally qualified to speak from a specific perspective—which is most of them—students are encouraged to explore these areas themselves. We believe we are all enriched by this.

We are always open to input such that the course can better serve these goals. Students should feel comfortable approaching us with their suggestions.

Office Hours

We will have limited availability outside of our class time, and won't keep scheduled "office hours." Students should not expect us to immediately solve specific design or technical problems, or have their progress be blocked by this. Their first resource should be themselves, then our course site and its materials, and then each other.

That said: if there are still questions—particularly logistical or content ones—students can message us any time on Slack, and we will respond when we can. But again, this should never be a bottleneck; all of this works better when not done at the last minute.

Additional Technical Help

For more specific technical instruction and questions, Parsons has dedicated CD-program tutors available to help students with HTML, CSS, and JavaScript—as well as offering general design critiques and feedback. Their daily schedules are available midway through the semester; sessions are by-appointment.

The University Learning Center also offers its own tutoring sessions on a wide range of subjects, beyond design and programming; these are also by-appointment.

As tutors are only available a limited number of hours per week, it is advisable to start early on your projects and seek help along the way—to avoid the usual end-of-semester rush for additional help.

Recording Sessions

We will take recordings of all of our class sessions for students to reference later. As these will include the students and their work, the recordings will be stored on our [Google Drive](#) and made available only to New School email users.

Plagiarism and Derivative Work

While all design work is, to some extent, built on the the continuum of culture before us—our expectation is that the students’ work in our class is *their own creation*. That is why we are all here, and there is no point in us engaging with any work that is not.

We have zero tolerance for plagiarism and passing off the work of others (or [LLMs](#)) as one’s own—from initial ideas through to final implementation. This ranges from “verbatim” (copying-and-pasting), to “paraphrasing” (rewording, rearranging, deriving), to “autocompleting” (with LLM-assisted editors), to “vibe coding” (wholesale agentic use). None of these are your creation.

Everything that is turned in should be a product of each students’ own ideas, synthesis, and execution. They will always be expected to understand it, explain it, and defend it as they would themselves.

Students should also review the school’s [Academic Integrity Policy](#).

Attribution

When referencing or incorporating materials outside of our course-provided ones (lectures, exercises, demos, recordings, Slack conversations, etc.), students should always attribute their use.

These attributions should explain *why* they were brought in, *what* was referenced, and an overall understanding of *how* it works. (This includes anything from LLMs.) This will both help us gauge what students are getting out of it, and also where we can expand and guide our learning together as a class.

LLMs and “Artificial Intelligence”

There has been much discussion and development in our field (and others) around large language models, a.k.a. “artificial intelligence.” Tools like the conspicuous ChatGPT, Claude Code, Cursor’s IDE, Google Gemini, and GitHub Copilot are upending the industry and changing how we work. But aside from (the many) other concerns, their use in the classroom remains fraught, at best.

Code Is Writing; Writing Is Thinking

Every course has different learning objectives. In ours, we think students first need to write code themselves in order to truly understand our medium. We believe that writing is thinking, and code is no different—the slowness, friction, and resistance of it are necessary for learning that lasts. We might in some ways get “less” done, but we think we will all take more from it.

We think that avoiding these difficulties is ultimately a disservice to your education. LLMs are the same as any other tool in this regard, and are always to be treated like any at our disposal—as *aides* to our understanding, not *shortcuts* around learning.

Students should also review our evolving program “AI” guidelines. Our class-specific LLM/“artificial intelligence” guidelines will live on our course site.

Attendance, Grading, and Other Policies

All CD classes adhere to the same common program and university policies.

Acknowledgments

We’d like to thank Lynn Kiang, Brendan Griffiths, Andrew LeClair, and the extended MPS CD family for their support in the planning and running of this course. And thank you, for reading this far.