

Welcome to INFO6150

INFO6150: Web UX Engineering (UI/UX)

DESIGNER



What my friends think I do.



What my mom thinks I do.



What I think I do.



What society thinks I do.



What I really do.

Quick Questions

- Is this an online class?
 - No
- Is in-person attendance required
 - Yes (except for excused absences)
 - Not my decision
 - University can be SERIOUS about this!
 - Let me know if sick or other excuse
 - I'm nice, we're all adults

Livecast Attendance - Boston/Silicon Valley

- NEU has legal needs for physical attendance
- You are required to physically attend almost every class you aren't excused from
 - Otherwise you fail the course
 - Not my rules
- If requesting an excuse
 - *I* trust you
 - Document because I'm not the only concern

Previous Experience

This course typically has both:

- People new to coding entirely
 - Often switching fields
- People with professional web dev experience

I try to help both groups

- Please be attentive and patient
 - Particularly in first few weeks

COVID Overview

- COVID sucks
- My rules are the same even without COVID
 - Being sick and spreading ANY illness is bad
- Thank you for keeping yourselves and others safe
- I will assist with this whenever possible

Vulnerabilities

- I have multiple comorbidities
- My wife has reduced immune system
- Expect due caution from myself
 - I extend the same courtesies to you

Who is this guy?

Brett Ritter [<b.ritter@northeastern.edu>](mailto:b.ritter@northeastern.edu)

- He/Him
- Currently in Seattle
- WebDev since 1995 (not designer)
- Multiple languages, frameworks, platforms
- Both frontend and backend
- Part time instructor since 2017
 - Tell me where to improve my teaching

Not Perfect

I have a truly terrible memory.

Terrible

You have my permission to remind me, and keep reminding me, until something is done or I explicitly say "stop".

Funny

I tell jokes

Fortunately, they are all hilarious

You will laugh out loud

Try it now

Get Better at Laughing

We will keep practicing that

What is UX Engineering?

- Newer term
 - Still being defined in practice
- Front End Web Developer is vague
 - More Coding?
 - More Design?
- "Designer" doesn't always mean ANY web skills
- Front End Web Dev + Designer = UX Engineering
 - NOT the skill sets of both
 - The blurry area in-between the two

Will you have a Designer?

General situation:

- Smaller shops (coding workplaces) = no designer
- Larger shops will have designers

Always better off with a good designer

- But often have to "make do" without

Limitation: I am not a designer

- I do not have design skills or experience
- I DO have experience working with designers
 - On real teams in different industries
- I will teach you how to work with designers
 - What to consider when you don't have one
 - What designers often overlook
- How to create different web interactions (UI)
 - How to decide if **helpful** interaction (UX)
- NEU offers separate design courses

The Web is Awesome

- Literally changed the world
- Socializing, Dating, Reference, Commerce
- Based in programming
 - Zero copy cost!
 - What other industry has that?
 - Build tools to save repeat effort!
 - Benefit from tools others write!

Coding Web UI/UX is Awesome

- Every website requires discovering how it works
 - Humans are naturally curious
 - Humans are also impatient
- Learning good UX is learning how humans work
 - Creating good UX is immensely rewarding
- Web UI/UX makes good functionality FEEL good
 - Power + satisfaction
- EVERYONE can run your code!
 - Does not require installation
 - Does not care about Operating System
 - Does not care about mobile/desktop

Web UI/UX evolves quickly!

- <https://www.webdesignmuseum.org/gallery/apple-1996>
- <https://www.webdesignmuseum.org/gallery/twitter-2006>
- <https://www.webdesignmuseum.org/gallery/netflix-2002>

What does INF06150 teach?

- How to implement UI/UX decisions
 - Using HTML, CSS, JS
 - Native "vanilla" JS
 - Intro to React
- Important needs in UI/UX
 - Accessibility (a11y)
 - Semantics
 - Maintainability
 - Avoiding Deceptive Patterns

What does INFO 6150 NOT teach

- Design
 - I cover more HOW than WHY
 - Taught from Web Dev perspective
- Server-side/Backend logic
 - See INFO 6250 for this
- Advanced Front End logic
 - See my INFO 6250 for this
 - Note: other Boston INFO 6250 backend only!
- Using Libraries (mostly)
 - Libraries are great!
 - But we are training *your* skills

What to expect

- Git
- Lots of HTML/CSS
- Focus on best practices
 - NOT "it works"
 - NOT "looks right"
- Many syntaxes
 - HTML
 - CSS
 - JS
 - React

How this course flows

- Weekly lectures
- Weekly assignments
 - Submitted via Github
 - Due night before next class
- Weekly Quizzes
 - On Canvas
 - Open Notes
 - Not time-limited
 - Due night before next class
- Projects (solo)
 - Like big assignments

How the Course is Graded

- Quizzes (10% of grade, lowest dropped)
- Assignments (15% of grade, lowest dropped)
- Projects (each 25% of grade)

Final Project can raise low grades!

Instructor Virtual Office Hours

- Mon 2pm-3pm (ET) / 11am-noon (PT)
- Tue 2pm-3pm (ET) / 11am-noon (PT)
- Wed (no office hours)
- Thu 2pm-3pm (ET) / 11am-noon (PT)
- Fri 2pm-3pm (ET) / 11am-noon (PT)
- Other times by appointment
- Available on Slack for quick questions

Teaching Assistant(s)

- Have taken this course!
 - Use their knowledge and experience
- Virtual Office Hours
 - To be announced

Extensions

I am reasonably generous with extensions

- Real Life happens
- No excuses needed
- No penalty for granted extensions
- Builds Real Job Skills

BUT!

- **Ask BEFORE day assignment is due**
 - Unpredictable problems are fine
- Don't overuse
- On-time grading gets priority before extensions

DO NOT COPY WORK

- I prefer learning to grades
 - But grades should be fair
- Most learning is practice
 - Finding little lessons
- Copying reduces practice
 - Whatever you call it ("referencing")
- NOT WORTH THE RISK
 - Use my generous extension policies
- See "do-not-copy-work" in your repository

Large Language Models (ChatGPT etc)

- Don't want to be old/out-of-touch
 - But want to teach
- LLMs are NOT "AI"
 - No "understanding"
 - Just predictive text
- Might be helpful on job (maybe)
 - NOT helpful to learn!
 - Like copying, cuts practice
 - Too often it is WRONG
 - You lack context to know

Tools for the Course

I have built this course to use tools like a job would

- I will note exceptions

Canvas

- Zoom Meetings link
- Quizzes
- Grades
 - Can take up to a week after due date
 - Details will be in Github
- Class Recordings
 - Zoom Meetings -> Cloud Recordings
 - Check the date/time!
 - Records any time someone connects
 - School requires physical attendance!

Zoom

- Your personal video is optional
- Classes are streamed and recorded
 - Recordings available in Canvas
- Virtual Office Hours
 - Link posted in Slack

Operating System

Mac, Windows, *nix

- All allowed for course (tools work on all)
- All exist in jobs
- Mac common (for devs) in bigger teams/companies
- Windows common in small teams/companies
- Windows common in explicit MS stack (.NET)
- *nix never common, but is underlying tech

I use Mac

- Windows users may need to adjust (rare)

Browser

A web browser is a significant tool for...web

- Course will use Chrome browser
- Current most common with users
- Many Devs prefer Firefox for development
- Our techniques should work in all major browsers
 - Developer Tools (DevTools) minor differences
- Real work should always be tested in all major
- But course will use only Chrome

Editor / IDE

- Used to edit HTML/CSS/JS/JSX/Markdown files
- Course does not have any requirement
- Instructor will use **vim** and **VSCode**
 - Instructor is old like dirt
 - Also vim presents fewer distractions
- **VSCode** is recommended for students
 - But not required
 - Most common with devs

Slack

We use **Slack** as our in-class chat

- Matches more jobs
- Actual job SKILL
 - Searching
 - Bouncing between channels
 - Not missing replies
 - Not starting with "Hi" (<https://nohello.com>)
- Sign up here: <https://rebrand.ly/seainfo6150-slack>
 - I need to unlock existing accounts
 - If you started the class previously

Slack Notes

- Good for code snippets
 - Use ` (backtick) around a command
 - Use ``` (triple backtick) around code block
- Screen shots often problematic
 - Unreadable on mobile
 - Can't copy code to test
- I am very available on Slack!
 - I'll respond when available
 - If you don't hear back, remind me!
- I announce changes to assignments, class cancellations, etc

Git

git is a version control system

- Tracks changes to files
- HEAVILY used in jobs
 - When not git, will be some other VCS
- Git is a lot to take in!
 - Notes will have suggestions to learn it better
- Being able to use git great for jobs
- Knowing more than the minimum even better!
- You will need to download/install

Github

- `git` manages a local (your computer) "repository"
- Github is a central repository ("repo")
- Each student will have their own Github repo
 - Get yours here: **<https://rebrand.ly/seainfo6150-github>**
- I push notes and assignments to github
- You pull to your local repo from github
- You push work from your local repo to github
- TAs and I grade and merge (more detail later)
 - Real job: You merge

NodeJS and related tools (npm, npx)

- Javascript (JS) for server-side
- We use it but not write for it
 - That's my 6250 class
- You will need to download and install it
 - Either "Latest Stable" or "LTS" version
 - Do not install node libs/packages as "sudo"/administrator

Class Recordings

- I cast and record classes on Zoom
 - Links available on Canvas
- Always a risk of technical problems!
 - Missing class is a risk
- University has physical attendance policy
 - Remember to get excused absences!

Accommodations

If you benefit from accommodations

- Lasting or temporary
- For reasons physical, mental, or emotional
- Let me know and I will work with you
- Your reasons can stay private

Your Repository

Key things to find in your repo

- Syllabus (`/syllabus.md`)
- Do Not Copy policy (`/do-not-copy-work.md`)
- Readings (`/readings`)
- Samples (`/samples`)
- Class Notes (`/classes`)
- Class Assignments (`/work`)
- Projects (`/project1`, `/project2`, `/final`)

Tips for Excelling in this course

- I teach more than is required to pass with an A
- What I teach should help you on the job
- Ask questions!
- Use DevTools to debug/diagnose!
- Do the assignments from scratch
 - NOT using others' code "as a reference"
 - Do NOT copy code
 - Practice teaches MORE than main lesson
 - Not relying on ChatGPT/LLMs
- Start Projects ASAP
 - Coding always takes longer than expected

If you fall behind...

- Tell me
 - As soon as you notice
- We can figure out how to fix things
 - "Working harder" is a nice thought
 - Often that's not the problem
- This is a JOB SKILL
 - Manager and team want to know ASAP
 - So they can prevent problems
 - Not about blame

Use Caution with Online Resources

- A lot in Webdev has changed over time!
 - Lots of outdated/misleading resources
- Never trust any source older than 3 years old
- Prefer **MDN** as a source
 - W3Schools is "okay"
 - I recommend MDN first
- **Never copy/paste code** (HTML/CSS/JS)
 - Understand and recreate it first
 - This includes ChatGPT
 - Copying limits your skill