

# Welcome to INFO6150

INFO6150: Web Design/User Experience



# Pandemics

We live in a time of large scale health concerns

- School requirements are defined
- I can support you
- If you need excused absences or remote attendance
  - Let me know

# About The Instructor

- Brett Ritter <[\*\*b.ritter@northeastern.edu\*\*](mailto:b.ritter@northeastern.edu)>
- Web Developer for 25+ years
- Teaching since 2017
  - Part time, not trained as teacher
- Terrible memory
  - Remind me of things
- Web Dev, not Designer
- Terrible memory

# What does INFO6150 teach?

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

# What does INFO6150 NOT teach

- Design
  - More HOW than WHY
  - Taught from Web Dev perspective
- Server-side/Backend logic
  - See INFO6250 for this
- Advanced Front End logic
  - See INFO6250 for this
- Using Libraries (mostly)
  - Libraries are great!
  - But we are training *your* skills

# What to expect

- 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
  - 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)
- 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 Assistants

- 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 late

# **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
  - Note: grading can take up to a week
  - Details will be in Github

Otherwise don't make much use of Canvas

- Let me know if you think I should!

# Zoom

- Classes are streamed and recorded
  - Recordings available in Canvas
- Virtual Office Hours
  - Link posted in Slack

## NOT USED/BARELY USED

- Zoom Chat (use Slack instead)
- Break out rooms (too busy)
- Your Video (You can, not required)

# Operating System

Mac, Windows, \*nix

- All allowed for course (tools work on all)
- All exist in jobs
- Mac common 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
- But course will use only Chrome



# Editor / IDE

- Used to edit HTML/CSS/JS/JSX files
- Course does not have any requirement
- Instructor will use **vim**
  - 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/svinfo6150-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
- I am very available on Slack!
  - Reach out anytime!
    - If I'm not available, I won't respond
  - 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
  - Even 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
- Each student will have their own Github repo
  - Get yours here: **<https://rebrand.ly/svinfo6150-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

# **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
  - I pay attention if you are doing poorly

# **Accommodations**

If you require any accommodations

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



# 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

Key tips:

- Ask questions!
- Do the assignments from scratch
  - NOT using others' code "as a reference"
    - Do not copy code
  - Practice teaches MORE than main lesson
- 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 over 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