PUCD 2026 ~ Section E
Core 2: Interaction Lab

Program	School of Art, Media, and Technology: Communication Design
CRN	2573
Semester	Spring 2021
Meeting Day	Friday
Meeting Time	9:00 – 11:40am
Building/Room/ Zoom	https://NewSchool.zoom.us/j/98580083262
Instructor & Email	Mark Beasley / beasleym@newschool.edu
Class Website	https://tidepool.school

Course Description
This course serves as a complement to Core Studio Interaction. The assignments are built to work in tandem with the projects students are developing in the studio class. The lab is designed around a series of small workshops that teach beginning and intermediate interaction design through a hands-on engagement with HTML and CSS.

Accessible Tasks
Lab Demos
Pre-recorded technical demos will be provided by Lab instructors to support the studio projects. Students are expected to watch these demos in advance of class for class discussion and in-class exercises.

Assignments
There will be homework assignments due by the start of class the following week and posted to your Github Page. These assignments will focus on a specific computing topic relating to the technical material requirements to the projects being assigned in Core Studio. Assignments will be introduced in class and listed in detail on our class site.

In-Class Exercises
Students will be expected to complete individual/group in-class exercises to push your knowledge of the concepts discussed in class.

Skills Test
A final skills test will be given on May 7 to evaluate your comprehension of the technical lab material.

Course Outline

Unit 1: Interaction with the Internet

In weeks 1-3 we will focus on:

- File management (naming, organization, file paths)
- Setting up and starting a new project
- Tools (code editor, inspector, Git/GitHub, Glitch)
- Local web development environment
- Review of HTML/CSS basic concepts and syntax
- Embedded content

Unit 2: Interaction with Type

In weeks 3-7 we will focus on:

- Digital typography and web fonts
- Responsive web design
- Multi-page navigation
- Creating structure with HTML and CSS
- CSS Layout + Style

Unit 3: Interaction with Data

In weeks 7–12 we will focus on:

- Real-time data sources (APIs/CMS)
- Javascript: the DOM and interacting with data
- AirTable as database and API

Unit 4: Interaction with People

In weeks 12–15 we will focus on:

- Creating dynamic outputs with HTML/CSS/JS
- Prototyping software

Learning Outcomes

By the end of the semester, students will be able to:

1. Apply skills in HTML

a. Standards: W3C, the World Wide Web Consortium: W3C recommendations as standards
b. Understand the difference between programming and markup
c. Title, Meta (keywords/descriptions); !DOCTYPE and Document Type Definition
d. HTML tags and the HTML Element Syntax including opening/closing tags, nested structures.
e. HTML Attributes: class, id, style, title
f. Headings, Paragraphs and Formatting
g. Links, lists, forms and images
h. The Box Model
i. HTML5 Semantic/Structural elements
j. HTML5 Media Elements

2. Apply skills in CSS

a. Cascading Style Sheets, their storage in external CSS files and reference in HTML
b. Styling backgrounds, text, links, lists and forms

c. Styling the Box Model: border, outline, margin, padding
d. Working with dimensions, positioning, display, floating and align
e. Color systems
f. Manipulating images with CSS
g. Using Webfonts
h. Using CSS to create interactive elements
i. Media queries and responsive design

3. Understand the meaning of JavaScript

a. Basic idea of JavaScript

4. Prepare Images for the Web

a. Digital Image Formats - what are they for and how are they created professionally: GIF, JPG, PNG
b. Working for different resolutions

5. Understand Web Environments

a. Getting it online: Purchasing URL/Webspace
b. Use FTP to upload files
c. Use in-browser tools to troubleshoot and amend HTML/CSS
d. Search engine optimization
e. The role of content management systems / blog systems

Materials and Supplies

Laptop

Technologies

https://tidepool.school
Classroom management for schedule, projects, 1:1 signups, and presentations

Slack
Class communication for all Core 2 and direct messaging with instructor and peers.

Git/GitHub
We will be using GitHub to manage our code. Sign up if you don't have an account already.

Atom
Atom is a sophisticated text editor for code, markup and prose. We'll also utilize the TeleType plugin for collaborative editing and debugging.

Glitch
Glitch is a real-time, collaborative coding platform that we'll use for in-class demos and exercises.

Google Chrome
A fast, secure, and free web browser. We'll be using Chrome exclusively in the class.

Vimeo
Watch lectures and demos on the **CD Vimeo account**. Password for all videos: **interaction**

Assessment Criteria

50% Assignments
20% In-Class Exercises
10% Core 2 Skills Test
20% Class Participation

Grading Rubric

The following grading rubric will be used to assess both mid-semester score and final grade.

Assignments / 50 pts

Assignment 1	# /10
Assignment 2	# /10
Assignment 3	# /10
Assignment 4	# /10
Assignment 5	# /10

In-Class Exercises / 20 pts

Delivery and completion of all exercises (5 pts per exercise)	# /20

Tests / 10 pts

Core 2 Skills Test	# / 10

Participation / 20 pts

Frequency Regular contribution to other's critique, enthusiasm, and quantity of comments	# / 10
Professionalism Timely and productive weekly progress, attitude to feedback, verbal and visual presentation of process and final work	# / 10

Total Score **# / 100**

A (95 – 100); **A-** (90 – 94); **B+** (86 – 89); **B** (84 – 85); **B-** (80 – 83);
C+ (76 – 79); **C** (74 – 75); **C-** (70 – 73); **D** (60 – 69); **F** (59 and below)

Attendance, Grading and Work Submission Standards, Program Policies, Making Resources, and University Policies

All CD classes adhere to the same program and university policies:
https://bit.ly/2LHztsW

Attendance Policy: For classes meeting once a week, students are allowed **2 absences**. Any absence beyond the allowed absences will result in an automatic failure (F) for the course. There are no excused absences, and doctor's notes are not necessary.

A student is deemed tardy if a student fails to arrive within 15 minutes past the beginning of class. **2 tardies** will result in an automatic absence. A student who arrives an hour past the beginning of class will be deemed absent.

CD Code Tutors

CD student code tutoring sessions are offered Monday, Wednesdays, and Fridays throughout the semester on a drop-in basis. Students are encouraged to supplement their Lab instruction with tutors to assist you on Studio and Lab assignments. The most updated schedule and Zoom meeting links are here: https://bit.ly/3rBDcso

Schedule

Schedule is subject to change per instructor discretion.

Unit 1: Interaction with the Internet

Week 1: Skills Review	
Synchronous	• Core 2 Lab Kickoff • Class introduction • Lecture: HTML Recap • Core 1 Skills Assessment
Asynchronous Demos	• HTML & CSS Basics • Embedded Content • Glitch
Homework	Start a personal profile for the class

Week 2: Embedded Content	
Synchronous	• Lecture: CSS Recap • Add class profile to Github
Asynchronous Demos	• GitHub & GitHub Pages • Local Web Development Environment

Homework	Make a playlist with a service or method of your choosing (mp3, youtube, spotify, itunes, etc) and embed it within a new page linked from your profile. Style the page with css in a way that compliments your playlist.

Unit 2: Interaction with Type

Week 3: Typography on the Web	
Synchronous	• Lecture: Font Resources and Services
Asynchronous Demo	• Typography on the Web
Homework	Add typographic treatment to your class profile and playlist page

Week 4: Responsive Design and CSS	
Synchronous	• Lecture: Responsive CSS Recap • Workshop: CSS animation
Asynchronous Demos	• CSS Layout + Style • Responsive Web Design
Homework	Assignment 1: Wikipedia clone

Week 5: Structure	
Synchronous	• Review assignment 1 <u>Small Group</u> • Technical Meetings: Studio Project 2
Asynchronous Demos	• Creating structure with HTML and CSS
Homework	Assignment 2: Wonder Cabinets

Week 6: Studio Project 2 Technical Meetings	
Synchronous	• Review assignment 2 <u>1:1 Meetings</u> • Technical Meetings: Studio Project 2
Asynchronous Demos	

Homework	

Unit 3: Interaction with Data

Week 7: Javascript Introduction	
Synchronous	• Lecture: Javascript Recap
Asynchronous Demos	• AirTable • Javascript: Getting Started
Homework	Assignment 3: Instruction Sets

Week 8: Mid-Semester Reviews	
Synchronous	• Review Assignment 3 1:1 Meetings with Instructor • Mid-Semester Evaluation
Asynchronous Demos	JavaScript + the DOM
Homework	

Week 9: API + Airtable	
Synchronous	Workshop: Web APIs
Asynchronous Demos	• API Overview • Creating an API key in Airtable
Homework	Assignment 4: Biomimicry

Week 10: Javascript + Data	
Synchronous	• Review Assignment 4 • Workshop: Javascript + data Small Group • Technical Meetings: Studio Project 3
Asynchronous Demos	Javascript + Data
Homework	

Week 11: Studio Project 3 Technical Meetings	
Synchronous	1:1 Meetings • Technical Meetings: Studio Project 3
Asynchronous Demos	
Homework	

Unit 4: Interaction with People

Week 12: Prototyping	
Synchronous	• Workshop: Javascript Events
Asynchronous Demos	• Prototyping in Software • Input/Output Examples
Homework	Assignment 5: Variation Machines

Week 13: Studio Project 4 Technical Meetings	
Synchronous	• Review Assignment 5 Small Group • Technical Meetings: Studio Project 4
Asynchronous Demos	
Homework	

Week 14: Studio Project 4 Technical Meetings	
Synchronous	1:1 Meetings • Technical Meetings: Studio Project 4
Asynchronous	
Homework	Core 2 Skills test prep

Week 15: Core 2 Skills Test	
Synchronous	Lecture: Putting sites online, documenting your work All Class

	• Class Discussion: Reflection and Q&A
Asynchronous	Core 2 Skills Test
Homework	Course evaluation