THE NEW SCHOOL
PARSONS
DESIGN+TECHNOLOGY

Parsons MFADT
Creativity and Computation Lecture/Lab
PGTE 5251-A CRN:6201
Syllabus, Fall 2015 (subject to change)

Lecture
Friday 3:50pm-5:10pm**, 63 5th Ave (University Center), room L104 (lower level)**
Faculty: **Sven Travis** (traviss@newschool.edu or sven@newschool.edu)
Lecture website: **my.newschool.edu and Canvas**
Lab website: **GitHub**

Sven's office is D12, 12th floor, 6 E 16th St. (meaning he doesn't have one)
Sven's office hours are by appointment,

Labs (Faculty and section information):
Julie Huynh (julie@newschool.edu)
CRN: 3525 - PGTE 5250 – A, Tuesday 7:00 pm - 9:40 pm, 6 East 16th Street 609

Pierce Wolcott (pierce@newschool.edu)
CRN: 3526 - PGTE 5250 – B. Tuesday 7:00 pm - 9:40 pm, 6 East 16th Street 1208

Madhav Tankha (tankm304@newschool.edu)
CRN: 3527 - PGTE 5250 – C, Tuesday 7:00 pm - 9:40 pm, 6 East 16th Street 1206

Gabor Tankovics (tankb969@newschool.edu)
CRN: 3531 - PGTE 5250 – D, Tuesday 7:00 pm - 9:40 pm, 6 East 16th Street 1200

Niki Selken (nicoleselken@newschool.edu)
CRN: 3698 - PGTE 5250 – E, Thursday 7:00 pm - 9:40 pm, 6 East 16th Street 601

Nicole Messier (messier@newschool.edu)
CRN: 3699 - PGTE 5250 – F, Monday 7:00 pm - 9:40 pm, 6 East 16th Street 1208

Alexander McClure (mccla636@newschool.edu)
CRN: 5404 - PGTE 5250 - G Thursday 7:00 pm - 9:40 pm, 6 East 16th Street 600

This is a required lecture/lab course for first year MFA Design and Technology students. Each student should be enrolled in the lecture and one lab section.

General Course Description
The primary intent of this course is to get all of us (students, faculty) to think about ourselves as creative individuals, and to investigate our relationship with digital technology, specifically coding and making. How do we become authors using computation and devices? The course will combine informational lectures that will touch upon a variety of topics with hands-on labs that will require the students to code and build projects related to specific technologies (Javascript, Arduino, openFrameworks, and others).

Lectures will vary between historical overview, scientific explanation, and philosophy. Although carrying on class discussion in such a big group can be a challenge, we will give it a shot. Hopefully ideas raised in this class will find their way to the studio, and vice-versa. Labs will be hands on. Each lab session will involve building code or devices to accomplish a specific goal. Students will be expected to carry out each lab module, and to apply them to projects. Three projects will be required in the lab during the semester (along with weekly homework), with a fourth project possible at the end of the semester.

This course will have two basic components: the weekly lecture and the weekly lab. The lecture will last one hour 20 minutes per week, and the lab will be a full two hours 40 minutes. The course is broken down as follows:

1. **Lecture** (63 5th Ave (University Center), room L104 (lower level)-- full group, about 80 students). In most cases, lecture topics will be somewhat parallel topics and work occurring in the lab. The intent is to provide historical, cultural, and other context for the technical subjects taught in the lab. There will not always be a perfect connection between the lecture/lab, but we'll do our best.

2. **Lab** (locations vary, sections of about 12 students—see above). This year the lab will focus on programming with Javascript, Physical Computing with Arduino, and a brief introduction to the C++ programming language using openFrameworks. We will spend about a third of the time on each topic. There will be a final project in each topic. At the end of the semester, there will be an elective lab project (meaning non-required). We will also be introducing you to one or more micro computer platforms for use in your work (either Raspberry Pi or Intel Galileo, TBD).

Assignments

There are two types of assignments for this course: readings and writings for the lecture (see schedule), and three project deliverables for the CC Lab course (one for each of the below listed Lab parts). It is considered desirable to combine projects undertaken in CC Lab with your Major Studio projects. You will have the opportunity to execute a fourth, optional Lab project at the end of the semester. In addition to Lab projects, you will be expected to continue to improve technically (and to demonstrate it in your project solutions), via a variety of weekly homework assignments throughout the semester. You should spend appropriate time designing, developing, and executing each homework assignment and/or project. The course is divided loosely into three parts, with 2 or 3 homework assignments within each part. The following description provides general details of each part, and the learning outcomes that the related project should demonstrate. Specific details of each course section will be provided in class.

Part One: Javascript:

a) "Vanilla Javascript"
b) "JQuery"
c) "Accessing APIs"

Part Two: Arduino:

a) "Basic electronics, overview of Arduino"
b) "Serial connections, sensors and control"
c) "Programming and networking"

Part Three: openFrameworks:

a) "Coding with C++, working with IDEs"
b) "Basic oF: examples, setting up projects, project structure"
c) "Writing code: classes"

Required deliverables for CC Lecture/Lab:

1. Two papers for the lecture
    - *"My Philosophy of Technology*", due September 25th
    - *"Why Technology is Bad"*, due Thanksgiving day
2. Weekly homework assignments within the CC Lab (may vary between sections).
3. Three final lab projects (Javascript, Arduino, openFrameworks).
4. One example of applying CC tech to a Major Studio project
5. Attendance and class participation in labs and lecture.

Optional deliverable for CC Lecture/Lab: final project (extra credit)

Assessable Tasks and Learning Outcomes

By the successful completion of this course, students will be able to:

1. Demonstrate an understanding of the iterative making process as relates to code or physical computing, using incremental methods such as prototyping and testing to build toward more advanced work
2. Demonstrate an understanding of the basic technologies presented in the CC Lab (web scripting, electronics and physical computing, compiled programming), including which projects might be appropriate to apply specific technologies too
3. Demonstrate effective application of course technologies to projects within the MFADT Major Studio environment
4. Understand why the technologies presented are relevant to the current (and future) worlds of art and design, as well as our broader society
5. Understand and execute "writing" of code, as compared to cut-and-paste borrowing or "reading" of code.
6. Be able to archive and document technical work in a demonstrative and reflective manner for presentation and referencing
7. Demonstrate an ability to recognize the importance of innovation, creative thinking, risk-taking, and experimentation

CC Lecture/Lab topics

(Order, dates, and actual topics may change—please check the course syllabus [**my.newschool.edu**] for updates-- Sven will address the class for approximately 90 minutes each week. Any schedule, topic, assignment due dates, or other changes will be announced in lecture.)

Lecture date	**Lecture** (always takes place the previous Friday)	**Labs**
Aug 28th Labs: week of Aug 31th	Course overview • Lecture/Lab topics Javascript RB Radical Jam	Vanilla Javascript
Sept 4th Labs: week of Sept 7th	*William Gibson* • Imagined Reality (RB jam) *Oliver Sachs*	Libraries/JQuery

Sept 11th Labs: week of Sept 14th	Data and information, databases • Edward Snowden, Wikileaks, Anonymous	Accessing APIs with JS
Sept 18th Labs: week of Sept 21st	Web servers Intro to hardware platforms (Arduino, Raspberry Pi, Galileo) The DIY/Maker movement. *Jeremy Rifkin*	Javascript project
Sept 25th Labs: week of Sept 28th	**Video or guest lecturer** TBA *"My Philosophy of Technology" paper due*	Intro to Arduino and overview of the board • Basic electronics • Soldering • Components
Oct 2nd Labs: week of Oct 5th	Computer architecture Operating systems Programming Languages *Ray Kurzweil*	Serial connections Inputs and outputs, sensors and motors Programming the Arduino
Oct 9th Labs: week of Oct 12th:	Networks • How they work • Protocols • Big art projects • Community *Gabriella Coleman*	Spacebrew
Oct 16th Labs: week of Oct 19th	Open Source • Linux, Apache, openFrameworks The state of new media: applications of creative coding *Steven Johnson*	Arduino project
Oct 23rd Labs: week of Oct 26th	**Video lecture** (Sven in China) Creative Coding Programming Languages Algorithms and algorithm design *Vikram Chandra*	Catch-up week IDEs (XCode, Code::works) Downloading and installing oF
Oct 30th Labs: week of Nov 2nd	**Guest lecturer** (Sven in China) *Katherine Moriwaki:* Wearable tech • Examples • Available devices	Intro to C++ openFrameworks: • Running examples • Project management/structure
Nov 6th Labs: week of Nov 9th	Combined with Lab: Intro to C++ and openFrameworks. Cinder *Bill Joy, Malcom Gladwell*	openFrameworks: • Add-ins • Input/output
Nov 13th Labs: week of Nov 16th	AI and machine intelligence • Minsky • Kurzweil Art and AI	openFrameworks: • Classes

	Howard Gardner	

Nov 20th Labs: week of Nov 23rd	Building complicated things Project process • Software engineering • Usability • Scale and scaling *China Mieville*	Labs: **No Class (Thanksgiving)**
Nov 27th Labs: week of Nov 30th	**Lecture: No Class (Thanksgiving)** *"Why Technology is Bad" paper due*	openFrameworks project
Dec 4th Labs: week of Dec 7th	Augmented reality Virtual and imagined realities Fantasy and Sci-fi	Final project workshop
Dec 11th Labs: week of Dec 14th	Presentation of outstanding Lab projects	Final project presentations

References/Resources/Readings

Important: you should always back up your work to disk (or to thumb drive or to SDCard). Do not count on Internet access, and you should always carry an extra digital copy of assignments when you need to show them in class or turn them in. I will always have a hard disk available for you to upload assignments to—email may or may not work to hand things in.

Required Readings for CC Lecture
Sven will periodically distribute short readings or essays for the CC Lecture, via my.newschool.edu, Canvas, or email (TBD). Books that we will likely refer to are (I provide Amazon links). Note: you are not required to purchase these books, but by all means do if you wish (they are great to have). Almost all are available used, and also in ebook/mobi (Kindle) format:

Gibson, William: The Peripheral, http://www.amazon.com/dp/0399158448
Sacks, Oliver: An Anthropologist on Mars, http://www.amazon.com/dp/0679756973
Csikszentmihalyi, Mihaly: Flow: The Psychology of Optimal Experience, http://www.amazon.com/dp/0061339202
Coleman, Gabriella: Hacker, Hoaxer, Whistleblower, Spy: The Many Faces of Anonymous, http://www.amazon.com/dp/1781685835
Rifkin, Jeremy: The 3rd Industrial Revolution, http://www.amazon.com/dp/0230341977
Kurzweil, Ray: The Age of Spiritual Machines, http://www.amazon.com/dp/0140282025
Chandra, Vikram: Geek Sublime: The Beauty of Code, the Code of Beauty, http://www.amazon.com/dp/1555976859
Gardner, Howard: Frames of Mind: The Theory of Multiple Intelligences, http://www.amazon.com/dp/0465024335
Johnson, Steven: How We Got to Now: Six Innovations That Made the Modern World, http://www.amazon.com/dp/1594632960
Mieville, China: Perdido Street Station, http://www.amazon.com/dp/0345443020

Required Readings/Resources for CCLab
All required readings for lab (software tutorials, etc.) will be posted to the Lab **GitHub** site. For the Arduino section, students are required to purchase the following two kits from "16hertz":

1. http://www.16hertz.com/product/uno-r3-kit-ultimate-arduino-compatible/
2. http://www.16hertz.com/product/16hertz-led-kit/

If you are enrolled in a Physical Computing elective, you will likely be purchasing the same kits, so you are set (you do not need to buy two).

Useful URLs

http://www.codecademy.com/en/tracks/javascript

http://www.w3schools.com/js/

http://learn.jquery.com/

http://arduino.cc

http://openframeworks.cc/

http://www.raspberrypi.org/

http://www.intel.com/content/www/us/en/do-it-yourself/galileo-maker-quark-board.html?wapkw=galileo

Software/Hardware (used in labs)
--A text editor (of your choice)
--Javascript (open-source: available at no cost)
--Arduino Uno board and affiliated components (we will distribute required kit info within the first couple weeks of class)
--XCode IDE (Mac), ***Code::blocks, or Visual Studio*** IDEs (open-source or available at no cost)—which one we used will be announced in class
--Raspberry Pi boards: don't go out and purchase one yet—we will talk more about this in lecture

Other resources
You should also be aware of the following resources. We will discuss in-class those we will use or depend on.

Lynda.com guide on The New School Library page - The New School Libraries have purchased a site wide license that is available to all faculty and students at the New School. Lynda is an online learning platform with video tutorials in a number of disciplines: 3D, video, business, photography, web design, graphic design, and more. There are many other digital resources available at TNS Libraries—you should check them out.

Adobe is one of the best resources for Creative Cloud tutorials (Premiere, InDesign, Photoshop, etc). Many of the Adobe tutorial videos are also on Lynda.com

Creative Commons Search. Copyright accessible materials: searching on this site assures you that the material you are using in a project has a Creative Commons Copyright agreement attached to it.

Youtube and Vimeo – Very handy for uploading and presenting code sketches, especially oF stuff.

Google Drive – please familiarize yourself with this as you may need to use it.

Lab access, printing, and equipment checkout for students
Visit **The New School's Academic Technology site** for information. There is a great deal of equipment available to you, and many different labs/printers. The main thing is not to wait until the last minute to figure it all out.

Grading and Evaluation

Students' ability to meet the course's learning outcomes will be evaluated based on the following criteria:

- evidence of the ability to solve problems, both creative and technical;
- evidence of the understanding of the project assignments and course material;
- the correct use of materials and formats specified;
- quality of work as evidenced in in-class exercises, final projects, sketchbook exploration and the learning portfolio;
- participation in class and online;
- improvement in technical, creative, and problem solving abilities;
- attendance in class and the timely completion of projects.

Final Grade Calculation for CC Lecture/Lab

20% Attendance and participation (students who miss more than two lectures and two labs, or any combination of four absences, should consider themselves in trouble)
20% Javascript
20% Arduino
20% openFrameworks
10% Effective application of Lab tech into Major Studio project(s)
10% Lab (elective) final project
100% TOTAL
Note: you will be graded by your CC Lab faculty, with input from Sven

Grading Standards

A [4.0; 96–100%]
Work of exceptional quality, which often goes beyond the stated goals of the course

A- [3.7; 91 –95%]
Work of very high quality

B+ [3.3; 86–90%]
Work of high quality that indicates substantially higher than average abilities

B [3.0; 81–85%]
Very good work that satisfies the goals of the course

B- [2.7; 76–80%]
Good work

C+ [2.3; 71–75%]
Above-average work

C [2.0; 66–70%]
Average work that indicates an understanding of the course material; passable
Satisfactory completion of a course is considered to be a grade of C or higher.

C- [1.7; 61–65%]

Passing work but below good academic standing

F [0.0; 0–45%]
Failure, no credit

Grade of W
The grade of W may be issued by the Office of the Registrar to a student who officially withdraws from a course within the applicable deadline. There is no academic penalty, but the grade will appear on the student transcript. A grade of W may also be issued by an instructor to a graduate student (except at Parsons and Mannes) who has not completed course requirements nor arranged for an Incomplete.

Grade of WF
The grade of WF is issued by an instructor to a student (all undergraduates and all graduate students) who has not attended or not completed all required work in a course but did not officially withdraw before the withdrawal deadline. It differs from an "F," which would indicate that the student technically completed requirements but that the level of work did not qualify for a passing grade. The WF is equivalent to an F in calculating the grade point average (zero grade points), and no credit is awarded.

Grades of Incomplete
The grade of I, or temporary incomplete, may be granted to a student under unusual and extenuating circumstances, such as when the student's academic life is interrupted by a medical or personal emergency. This mark is not given automatically but only upon the student's request and at the discretion of the instructor. A Request for Incomplete form must be completed and signed by student and instructor. The time allowed for completion of the work and removal of the "I" mark will be set by the instructor with input from Parsons Office of Advising:

Divisional, Program and Class Policies

● Responsibility
Students are responsible for all assignments, even if they are absent. Late assignments, failure to complete the assignments for class discussion and/or critique, and lack of preparedness for in-class discussions, presentations and/or critiques will jeopardize your successful completion of this course.

● Participation
Class participation is an essential part of class and includes: keeping up with reading, assignments, projects, contributing meaningfully to class discussions, active participation in group work, and coming to class regularly and on time.

● Attendance
Faculty members may fail any student who is absent for a significant portion of class time. A significant portion of class time is defined as three absences for classes that meet once per week and four absences for classes that meet two or more times per week. During intensive summer sessions a significant portion of class time is defined as two absences. Lateness or early departure from class may also translate into one full absence.

● Blackboard or Canvas
Use of Blackboard may be an important resource for this class. Students should check it for announcements before coming to class each week.

● Delays
In rare instances, I may be delayed arriving to class. If I have not arrived by the time class is scheduled to start, you must wait a minimum of thirty minutes for my arrival. In the event that I will miss class entirely, a sign will be posted at the classroom indicating your assignment for the next class meeting.

● Academic Integrity

This is the university's Statement on Academic Integrity: "Plagiarism and cheating of any kind in the course of academic work will not be tolerated. Academic honesty includes accurate use of quotations, as well as appropriate and explicit citation of sources in instances of paraphrasing and describing ideas, or reporting on research findings or any aspect of the work of others (including that of instructors and other students). These standards of academic honesty and citation of sources apply to all forms of academic work (examinations, essays, theses, computer work, art and design work, oral presentations, and other projects)."

It is the responsibility of students to learn the procedures specific to their discipline for correctly and appropriately differentiating their own work from that of others. Compromising your academic integrity may lead to serious consequences, including (but not limited to) one or more of the following: failure of the assignment, failure of the course, academic warning, disciplinary probation, suspension from the university, or dismissal from the university.