

CUNY Tech Prep 2016-2017: Fall Semester Technical Curriculum

Subject to Change

Program Overview

The CUNY Tech Prep program empowers you, an exceptional CUNY Computer Science student, with in-demand software development and professional skills so you can land the kind of technical job you want.

Our technical curriculum focuses on full stack web development, from the design and architecture of applications, through their implementation, testing, and deployment. At each stage, the course will help you apply the tools, concepts, and processes—such as Git, testing, CI/CD, agile development, etc.—preferred by industry today. Meanwhile, CUNY Tech Prep’s two full time career coaches will meet with you one-on-one, guiding you through *only* the soft skills *you* need, from resume crafting to behavioral interviewing. Our industry partners will join us throughout the year to guest lecture on hot topics, offer insight into their business practices, and give feedback on students’ technical and professional development.

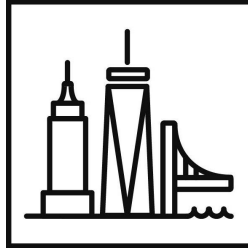
To learn the elements of full stack development, form a cohesive small development team, and build a successful demonstration project for prospective employers, it is imperative that you attend your section every week. Once a week, your section will meet for a two-hour class or a two-hour lab. Each class covers material identified and vetted by our Tech Industry Partners as central to competitive employment. Each lab is devoted to reviewing concepts and building a portfolio of applications with your teammates. We will help you transform what you’ve learned in the classroom into a portfolio that gives employers a clear and concrete sample of your potential. Our program emphasizes group assignments to prepare you for the team-centered development processes of leading firms. That’s also why we require you to deliver regular project updates and presentations; they keep you on track and strengthen your communication skills. CUNY Tech Prep will support your preparation for competitive tech jobs through coding challenges on HackerRank that revisit CS fundamentals and supplemental materials tailored to the technical aspects of the job search process, like the technical interview. Both Instructors and Teaching Assistants will have office hours to resolve outstanding questions, assist with projects, review class subject matter, etc.

Absences

- An unexcused absence is grounds for removal from the program.
- Multiple instances of arriving late and/or leaving early are equivalent to an unexcused absence.
- An absence can only be excused by your instructors and/ or program manager.
 - Under extenuating circumstances, notify your instructors and program manager beforehand.
 - Planned absences should be requested to the *program manager* at least 7 days in advance.

Projects and Homework

- Homework will be regularly assigned and must be completed by the due dates.
 - Failure to complete assignments is grounds for removal from the program.
- Progress on projects is expected at regular intervals.
 - Failure to show project progress is grounds for removal from the program.
- Group Projects
 - All group members are expected to make contributions.
 - Groups will be assessed both as teams and individuals.

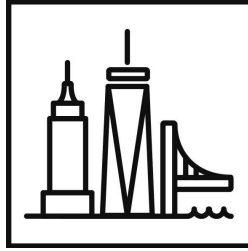


The Tech Stack curriculum

- Frontend Technologies
 - HTML + CSS (Bootstrap)
 - JavaScript [client-side] (jQuery)
 - CSS Preprocessors (Sass)
 - Templates (Handlebars)
 - React.js
- Backend Technologies
 - JavaScript [server-side] (Node + Express)
 - PostgreSQL
 - Java (Play Framework)
 - MongoDB
- Development Environment/Tooling
 - Linux OS (Vagrant VM)
 - Git (Github)
 - Heroku (or other cloud VM providers)

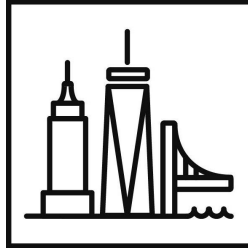
Full-Stack Topics: The frameworks we will use in this class are the Node+Express Framework and the Play Framework. The topics covered will be:

- MVC (Model-View-Controller) pattern
- Templating
- Routing
- ORM (Object Relational Mapper) / Data Mapper pattern
- Relational Databases, Basic SQL, and NoSQL Databases
- Testing (Unit, Integration, TDD, BDD) / Continuous Integration
- Application Deployment



Summer Pre-Work Curriculum

- Session 1
 - Introduction to the course
 - Environment Setup
 - Vagrant, Virtual Box
 - Linux OS basics
 - Environment: Ubuntu Vagrant VM
 - Common Bash commands, Apt-Get package management, Filesystem
 - Homework
 - Self-paced JavaScript class
 - Self-paced HTML/CSS class
 - JavaScript Quiz
- Session 2
 - HTML5/CSS Introduction/Review
 - Bootstrap Framework Introduction
 - Review JavaScript Quiz
 - Core JavaScript Overview
 - Homework
 - Create and Publish a Personal (Professional) Webpage
- Additional Summer Work
 - Self-Paced Java 8 class
 - Java 8 Quiz
 - Self-paced Git class
 - Self-paced Linux/BASH class
 - Self-paced Network basics/tools: IP's, ports, URL's
 - Self-paced JQuery class
 - JQuery Quiz



Program Schedule

(08/21 – 08/26) **Class 1** (please note that we start before the regular fall schedule)

In class:

- Java vs. JavaScript
 - Compiled vs Interpreted
 - Static/Dynamic Typing
 - Strong/Weak Typing
- Data Structures and Algorithms (Overview for Labs)
 - Arrays, Linked Lists, Trees
 - Stacks, Queues, Sets, Maps, Hash Maps
 - Searching, Sorting, Indexing

At home:

- Clone and run the first Full Stack Application (Microblog)
- Begin forming teams

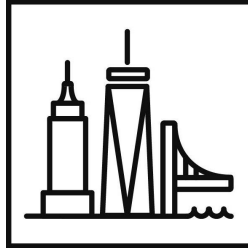
(08/29 – 09/02) **Class 2**

In class:

- Client-Server Communication
 - Network basics review (IP's, TLD's, DNS, Application Protocols, Ports)
 - Understanding HTTP Request – Response
 - Using CURL
- Designing a Full Stack App (Microblog)
 - Requirements Gathering
 - Screen Mockups (UI vs UX)
 - Responsive Design
 - Data Modeling
- Teardown of the Microblog code
 - Overview of major parts (MVC, DB, Tests)

At home:

- Use TDD to add Microblog User
 - Each team will propose two projects
-



(09/06 – 09/09) **Lab 1**

In lab:

- Data Structures
 - Arrays, Linked Lists
- Assignments Review
- Projects
 - Present project proposals, get audience feedback
 - Create mockups and plan data models

(09/12 – 09/16) **Class 3**

In class:

- Introduction to the Express.js Framework
- The Model-View-Controller (MVC) architecture
- Databases, Models, and CRUD (Create, Retrieve, Update, Delete)
- Building an app in Express.js Framework from scratch

(09/19 – 09/23) **Lab 2**

In lab:

- Data Structures and Algorithms
- Assignments Review
- Projects:
 - Teams start their projects. Begin writing tests. Post to Github.

(09/26 – 09/30) **Class 4**

In class:

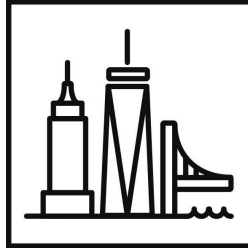
- Testing (TDD vs BDD, Unit/Integration)
- Code Reviews (Guest Lectures)
- CI/CD (Continuous Integration/Continuous Deployment)
- RESTful resources and routing (GET, POST, PUT, DELETE)

(10/05 – 10/07) **Lab 3a**

(10/13 – 10/14) **Lab 3b**

In lab:

- Data Structures and Algorithms
- Assignments Review
- Projects:
 - Teams pair up and do Code Reviews



(10/17 – 10/21) **Class 5**

In class:

- Sessions/Cookie Data
- User Authentication (*Security Topic*: Authentication vs Authorization)
- Controllers in Depth (*Security Topic*: User input - CSRF, Params)
- Views (Templates) in Depth (*Security Topic*: HTML Escaping)
 - Alternative HTML templating languages
 - Alternative outputs (JSON, XML)

(10/24 – 10/28) **Lab 4**

In lab:

- Data Structures and Algorithms
- Assignments Review
- Projects:
 - Implement User Authentication

(10/31 – 11/04) **Class 6**

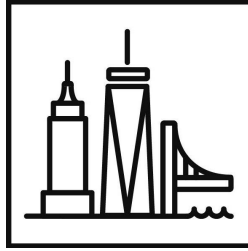
In class:

- Relational Databases
 - Entity-Relationship Modeling
 - Data Normalization
 - Relations (1-to-1, 1-to-Many, Many-to-Many) and Join Queries
 - Codd Rules
- Basic SQL (*Security Topic*: SQL Injection)
- Models in Depth (ORM vs Data Mapper vs SQL)
 - Where should business logic go?

(11/07 – 11/11) **Lab 5**

In lab:

- Data Structures and Algorithms
 - Assignments Review
 - Projects
 - Presentation Check-In / Practice
-



(11/14 – 11/18) **Class 7**

In class:

- API Concepts (vs RPC)
 - JSON,XML, and AJAX requests
- Introduction to Front-End Frameworks (React.js)
- Introduction to Real-Time Communication (WebSockets)
- Introduction to Micro-Services

(11/21 – 11/25) **No class: Thanksgiving Break**

(11/28 – 12/02) **Lab 6**

In lab:

- Data Structures and Algorithms
- Assignments Review
- Projects

(12/05 – 12/09) **Class 8**

In class:

- NoSQL Databases (Key-Value Stores and Document Stores)
- Building a Micro-Service
 - Java Play Framework
 - MongoDB
- Integrating a Micro-Service
 - REST API
 - Deployment to AWS

(12/12 – 12/16) **Lab 7**

In lab:

- Data Structures and Algorithms
- Assignments Review
- Projects:
 - Finish up projects for DEMO