CURRICULUM OVERVIEW

The digital revolution has transformed virtually every area of human activity—and you can be part of it as a web development professional. The Coding Boot Camp at Johns Hopkins Engineering is a part-time, 24-week Full Stack Flex course that gives you the knowledge and skills to build dynamic end-to-end web applications and become a full stack web developer.

Courses are scheduled to fit into your life, whether you’re employed or attending college full-time, with convenient weekend and evening sessions.

The program is rigorous and fast-paced and covers both the theory and application of web development. As you gain proficiency, you’ll use what you learn on real projects under the guidance of area practitioners. Plus, you’ll have an impressive Professional Portfolio and the confidence to succeed as a web development professional.
Are you creative, curious and looking to reinvent yourself professionally? If any of the following describes your situation—enrolling in our coding boot camp could be a smart career move:

- You’re considering a career change but not sure how to take the first step.
- You’re happy in your current field, but want to move to another company—or stay put but shift from a non-technical into a technical position.
- You want to engage more deeply with your current job—or boost your earnings and broaden your experience with freelance work.
- You have an entrepreneurial idea and need to acquire the skills to go “all in” on it and launch your business.
- You’re a full-time student but hungry to learn more and expand your skill set.
The **Skills** You’ll Gain

You will graduate with full stack web development skills*, including:

### Computer Science applied to JavaScript
- Algorithms (Searches, Sorts)
- Efficiency
- Time Complexity
- Big O Notation
- Data Structures

### Browser Based Technologies
- HTML
- CSS
- JavaScript
- jQuery
- Responsive Design
- Bootstrap
- Handlebars
- Local Storage, Session Storage, IndexedDB
- React.js

### Java

### Databases
- MySQL
- MongoDB

### Server Side Development
- Node.js
- Express
- User Authentication
- Progressive Web Applications (PWAs)
- MERN Stack (MongoDB, Express.js, React.js, Node.js)

### Quality Assurance
- Unit Testing
- Functional Testing
- Linting
- Continuous Integration

### API Interaction
- API
- JSON
- AJAX

### Deployment/Command-Line Fundamentals
- Heroku
- Git
- Github Pages

*The materials covered in this course are subject to change due to market demand.*
Building On The Basics

In web development, you can’t succeed without a solid grounding in the fundamentals. That’s why our curriculum begins with a deep dive into the basics of coding and data structure. That said, we recognize that the surest way to impress prospective employers and get job offers is to demonstrate your skills on real-world projects. You’ll have ample opportunity for hands-on involvement in outside projects, which will make up your Professional Portfolio.
Our graduates will have the opportunity to be placed in many different roles, including:

<table>
<thead>
<tr>
<th>Full Stack Developer</th>
<th>Software Developer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Front end Web Developer</td>
<td>Application Development Manager</td>
</tr>
<tr>
<td>Back end Web Developer</td>
<td>Computer Programmer</td>
</tr>
<tr>
<td>Product Manager</td>
<td>Web Designer</td>
</tr>
<tr>
<td>Technical Project Manager</td>
<td>Email Developer</td>
</tr>
<tr>
<td>QA and Test Engineer</td>
<td>Web Producer</td>
</tr>
<tr>
<td>Technical Business Analyst</td>
<td></td>
</tr>
</tbody>
</table>
By the time you graduate, you can expect to be able to:

<table>
<thead>
<tr>
<th>What You Will Learn</th>
</tr>
</thead>
</table>
| Apply “social coding” accepted and best practices (including source control, issue tracking, functional feedback, etc.) | Expertly navigate the file system and terminal basics  
| Build a front end website either from scratch or by utilizing a front end framework (such as Bootstrap) | Work independently or in a group on complex projects throughout the entire development lifecycle  
| Deploy static and dynamic websites to the cloud | Understand the basics of troubleshooting and enhancing legacy code  
| Implement complex logical conditions to meet an objective | Communicate the basics of serving a webpage and how the browser renders the code  
| Write SQL commands to perform Create, Read, Update and Delete commands | Create RESTful API’s utilizing JSON as a data format  
| Create a full stack Single Page Application with AJAX communication | Consume RESTful API’s properly utilizing REST verbs  
| Develop your vision for a website—and then build it! | Create web applications and services in Java  
| Create session-based applications utilizing user authentication schemes that are well-known and widely used |
Course Structure

Over the course of 24 weeks, you’ll attend informative lectures and take part in a variety of individual and team exercises, working independently and in groups, in the classroom and at home. Homework assignments provide an opportunity to apply what you’ve learned and build on it. The goal is to give you a comprehensive learning experience and true insight into a “day in the life” of a full stack developer.

- DISCUSSION
  Instructor-led discussions cover the background, history and use of a new technology or concept.

- LAB WORK
  You’ll put classroom teaching into practice individually and with a team to work on timed in-class exercises and projects.

- PORTFOLIO PROJECTS
  Your portfolio signals to employers that you are ready for primetime! You’ll build a substantial portfolio of projects that demonstrate your abilities across a wide variety of technologies.
We’re Here To Help

As you move up the learning curve, you’re likely to have questions around some of the concepts covered in class. We’re here to help—through in-person and virtual office hours, as well as a dedicated #slack channel where you can get assistance from instructors, support staff and your fellow students. All work is done via Github, so you can create issues directly on your own projects for instructors to assist you in a truly asynchronous fashion. In addition to learning to code, you will have access to career services that will help you prepare for technical roles after graduation such as:

Career Content and Practice Sessions

- Online Career Events With Industry Professionals
- Soft Skills Training
- One-on-One Career Coaching

Database of Customizable Tools and Templates

- Multiple Technical Resume Templates
- Github Best Practices
- Guidelines for Building a Portfolio
- Creating an Elevator Pitch
- Developing a Bio
Building Your Portfolio

It’s a fact: Companies care about what you can do, not what you say you can do. For that reason, our curriculum teaches you how to put what you’ve learned to work on actual portfolio projects, ranging from simple HTML and CSS code samples to sophisticated Single Page Applications with back end databases.
Building Your Portfolio

Your Full Stack Portfolio Page

Once you complete our program, your portfolio page will help you showcase your work with links and descriptions to the projects you’ve created, code samples, and personal information that employers want to see. Think of your portfolio page as your new home on the web.

Skills Needed

- HTML5
- CSS
- JavaScript
- Bootstrap
- Heroku
- Git

Objectives

- Create a home on the web to showcase your skills
- Build a complete site from concept
- Commit code to a shared repository

Business-Oriented Homework Projects

Our homework assignments are designed to emulate two real-world scenarios: 1) on-the-job tickets; and 2) job-seeking coding challenges. In both cases, the assignment is framed as a user story. In addition to user stories, ticket-based homework assignments follow the Agile project management conventions of framing the issue in terms of business context and acceptance criteria.

Skills Needed

All homework:
- HTML
- CSS
- JavaScript
- Git

Select homework:
- jQuery
- Bootstrap
- MySQL
- Node.js
- Express.js
- ORM
- API Consumption
- Heroku
- NoSQL
- React
- HTML5/CSS
- JavaScript/jQuery
- State Management

Objectives

- Each assignment focuses on a specific layer of the tech stack; objectives will vary based on the tech stack focus.
Self-Selected Front End Project
This is a group project that forces you to think outside your comfort zone. You and your group will decide what to build and then build it—a front end application that interacts with real-world services like Google Maps, Twitter and OMDB API.

Skills Needed
- HTML5/CSS
- JavaScript/jQuery
- API Consumption
- Bootstrap
- Git
- Heroku

Objectives
- Work in a group to build a project together
- Interact with third-party services
- Think in terms of mobile responsive design
- Read/write from/to a remote database

Full Stack Project
In your first full stack web application, you’ll create an intuitive front end/robust back end and scalable database.

Skills Needed
- HTML5/CSS
- Interactivity (AJAX)
- JavaScript/jQuery
- MySQL
- State Management
- Node.js
- Sessions
- Express.js
- Bootstrap
- ORM

Objectives
- Track issue progress with industry standard tools
- Communicate with team members asynchronously
- Design a MySQL Database Schema
- Create a full stack application
- Write project documentation
- Understand database relationships
Building Your Portfolio

Web Applications with Java

Learning Java will provide you with a firm foundation in one of the most popular and employable technologies both locally, and within the larger world of web development.

**Skills Needed**
- HTML/CSS
- Java
- Maven
- Git

**Objectives**
- Create a Java-based project
- Use Spring Data to build database-backed, dynamic applications
- Build RESTful APIs and Services
- Build a foundation in classical Object-Oriented Programming and Design in Java
- Develop familiarity with core J2EE APIs

Final Project

You will work independently or break out into groups to collaborate on a final project. You will come up with your own project and actually build it. The skills you learn during this project will truly help you to prepare for interviews and jobs.

**Skills Needed**
- Everything you’ve learned!

**Objectives**
- Define project scope
- Quality Assurance testing
- Responsive Design
- Deployment
- Code Organization
<table>
<thead>
<tr>
<th>Phase</th>
<th>Description</th>
<th>What You’ll Learn</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Phase 1:</strong></td>
<td>The first phase, Foundation, equips you with the fundamental concepts of web</td>
<td>› HTML, CSS, and JavaScript</td>
</tr>
<tr>
<td><strong>Foundation</strong></td>
<td>development, covering HTML, CSS, and JavaScript, as well as command line</td>
<td>› Creating a web page from scratch</td>
</tr>
<tr>
<td><em>(Weeks 1-8)</em></td>
<td>fundamentals and API consumption.</td>
<td>› Mastering terminal commands</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› DOM manipulation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› jQuery</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› Consuming RESTful APIs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› Parsing JSON to extract meaningful data</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› Using AJAX to update data on a website</td>
</tr>
<tr>
<td><strong>Phase 2:</strong></td>
<td>In the second phase, Technical, you learn the skills necessary to engineer</td>
<td>› Writing Node.js server code to serve static web pages</td>
</tr>
<tr>
<td><strong>Technical</strong></td>
<td>a full stack web application, working with servers, databases, and other</td>
<td>› Querying large amounts of data and answering questions from a MySQL database</td>
</tr>
<tr>
<td><em>(Weeks 9-16)</em></td>
<td>back end technologies, and connecting them to the front end.</td>
<td>› Understanding and using Joins, Wheres, and Counts strategically</td>
</tr>
<tr>
<td><strong>Phase 3:</strong></td>
<td>The last phase, Performance, has a dual meaning in that you acquire skills</td>
<td>› Utilizing NoSQL databases, such as MongoDB, as an alternative to MySQL</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>to optimize your web applications for speed and efficiency as well as</td>
<td>› Improving the performance of applications</td>
</tr>
<tr>
<td><em>(Weeks 17-24)</em></td>
<td>prepare yourself for the transition to a career in web development.</td>
<td>› Converting traditional applications into progressive web applications (PWAs)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› Creating single-page applications with React</td>
</tr>
<tr>
<td></td>
<td></td>
<td>› Computer Science applied to JavaScript (data structures, algorithms)</td>
</tr>
<tr>
<td><strong>Asynchronous Regional</strong></td>
<td>Java is a mature programming language trusted across the software industry</td>
<td>› Create scalable web apps, APIs, and Services</td>
</tr>
<tr>
<td><strong>Content</strong></td>
<td>to build safe, scalable, and robust applications.</td>
<td>› Take a deep dive into core Java and Object-Oriented Programming</td>
</tr>
<tr>
<td><em>(Week 25)</em></td>
<td></td>
<td>› Build a foundation in common build tools for Java projects, such as Maven</td>
</tr>
</tbody>
</table>