

# RUTGERS | CODING BOOTCAMP

## OVERVIEW: FULL-STACK FLEX PROGRAM

Rutgers Coding Bootcamp Full-Stack Flex Program covers frontend, backend and full-stack development skills in one comprehensive class. Our 24- week course is designed to accommodate your schedule, and it can transform hardworking students with no coding experience into entry-level, full-stack developers. This hands-on, project-based coding course teaches you to work in an agile environment as you develop the skills you need to build dynamic and versatile web applications.

### Why should you take this course?

- You want to change your career or shift from a non-technical to technical role in your company.
- You may want to take on freelance work to supplement your income or stay engaged with the work that you do.
- You might want to go “all in” on an entrepreneurial idea and need the skills to build your product.
- You are a current student hungry for additional knowledge or focused on enhancing your skills.

***Anyone that wants to challenge themselves and learn a highly valuable skill should take this course!***

This program can provide you with the skills, frameworks, and support needed to build web applications.

*You will graduate from our course with a wide range of web development skills, such as:*

- HTML
- CSS
- Git
- JavaScript
- jQuery
- Lodash
- Responsive Design
- Node.js
- Express
- Object Relation Mapping (Sequelize.js)
- Heroku
- Object Oriented Coding
- Database Schema Design
- Security and Session Storage
- User Authentication

## REAL WORLD SKILLS

The fastest way to start a new career in web development is by demonstrating your skills. For that reason, you do cover theory and foundations and then quickly dive into projects that provide real world experience employers seek in interviews. Our course centers around projects so that you can actually experience coding firsthand. Yes, we cover theory and foundations; more importantly, we ensure that you have the real world experience to hit the ground running. Our course centers around projects so that you can actually experience coding firsthand, rather than simply learning theories.

*This program can help prepare you for the following roles:*

- Full-Stack Developer
- Backend Web Developer
- Technical Project Manager
- Frontend Web Developer
- Product Manager
- QA and Test Engineer

## PROGRAM LEARNING OBJECTIVES

*By the time you graduate from this course, you should be able to:*

- Develop your vision for a website—and then build it!
- Navigate the file system and terminal basics.
- Work independently or in a group on complex projects throughout the entire development life cycle.
- Understand the basics of troubleshooting and enhancing legacy code.
- Communicate the basics of serving a webpage and how the browser renders code.
- Apply accepted “social coding” and its best practices, including source control, issue tracking, and functional feedback.
- Build a frontend website either from scratch or by utilizing a frontend framework (such as Bootstrap).
- Deploy static websites to a cloud provider.
- Create and deploy dynamic websites to a cloud provider utilizing a backend framework.
- Implement logical conditions to meet an objective.
- Write SQL commands to perform Create, Read, Update and Delete commands.
- Utilize an ORM to manage a database schema and connection from a server.
- Create a full-stack Single Page Application with AJAX communication.
- Create and consume RESTful APIs utilizing JSON as a data format.
- Create session based applications utilizing user authentication schemes that are well-known and widely used.

## COURSE STRUCTURE

Full-Stack Flex is a 24-week, 250-hour program designed around your busy schedule. There are three distinct yet synergistic components: lectures, labs and portfolio projects. Much like in the real world, you will work individually and in group settings. We aim to provide the very best in terms of career preparation, and for your part, you must be prepared to spend time outside of the classroom. Our course provides homework assignments to best maximize your ability to be a full-stack developer.

### Lectures

Each lecture is 80-90 minutes and will cover a great deal of content. Slides will be made available to you after they are presented.

### Lab Work

The labs are 80-90 minutes each and geared towards giving you the ability to demonstrate your mastery of the course material. Labs may be either individual or group based

### Portfolio Projects

To really impress future employers, Full Stack Flex guides you to build a healthy portfolio of projects that demonstrates your abilities across a wide variety of topics.

## ALWAYS THERE TO HELP

Learning something new takes time, and sometimes you need to workshop a question or concept. We are here to help! We have in-person office hours before class and a dedicated #slack channel so that you can get help from our instructors, our support staff and also each other. All of our work will be done via Github, so you can create issues directly on your own projects for our instructors to assist with in a truly asynchronous fashion.

## PORTFOLIO

Let's face it, employers REALLY care about what you've actually done—not just what you say you can do. If you are willing to invest the time, you'll finish this program with a portfolio of projects that employers are seeking—from simple HTML and CSS code samples to fully completed and deployed Single Page Applications with backend databases.

### Full Stack Flex Portfolio Page

Once you complete our program, your portfolio page will help you showcase your work. It will include links and descriptions to the projects you create, code samples, and personal information that employers want to see. Students will also create their very own Content Management System (CMS) and that's a HUGE PLUS for employers. Think of this as your new home on the web.

#### Skills Needed:

- HTML
- CSS
- JavaScript
- Git
- API/JSON
- Bootstrap

#### Objectives:

- Have a home on the web to showcase your skills.
- Build a complete site from scratch.
- Gain familiarity with issue tracking systems.
- Commit code to a shared repository.

## Connect Four

Building a game has many benefits, like interface design, state management and, of course, fun! This game covers state management, edge cases, and determining win paths. Sometimes, things that seem simple, like seeing if four pieces fit in a row, are deceptively complex. With this project, our students also learn intangible skills like how to best tackle a difficult problem.

---

### *Skills Needed:*

- HTML5/CSS
- JavaScript/jQuery
- State Management
- Bootstrap

### *Objectives:*

- Build a fully functional game.
- Track winning and losing stats.
- Apply logic skills to a real project.
- Understand the basics of iteration.

## Frontend Project

The frontend project is a group project that forces you to think outside your comfort zone. It's a frontend application that interacts with real-world services like Google Maps, Twitter, and movie databases.

---

### *Skills Needed:*

- HTML5/CSS
- JavaScript/jQuery
- RESTful API
- JSON
- Firebase

### *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.

## Rutgers Virtual Flyers

This is the first full-stack app that you will build. Walking around Rutgers, you can't help but notice flyers posted everywhere. We will be taking that concept to the web with **Rutgers Virtual Flyers!** Looking for a great place to eat? Post a Virtual Flyer asking for suggestions! Want to know the best place to park before the game? Post a Virtual Flyer!

---

### *Skills Needed:*

- HTML5/CSS
- JavaScript/jQuery
- State Management
- Sessions
- Bootstrap
- MySQL
- Node.js
- Express
- Sequelize
- AJAX
- Lodash

### *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.

## Data Scraper and Parser

Not all applications have sexy UIs and pizzaz. Some are a lot more functional. In this project, you will be building a scraper to parse through large datasets to find something of value. “Big data” isn’t just a phrase thrown around by data scientists anymore—now you can make your own contributions to the field.

---

### *Skills Needed:*

- Node.js
- Request
- MySQL
- Sequelize

### *Objectives:*

- Understand how to optimize iterations and loops.
- Identify and protect against edge cases.

## Final Project

You will be broken into groups to work on a final project. This is going to be something really interesting—we can feel it! You will have to come up with your own project and actually build it. The skills learned during this project will help you to prepare for your first interviews and jobs!

---

### *Skills Needed:*

- Everything we’ve learned!

### *Objectives:*

- Define project scope
- Quality Assurance testing
- Responsive Design
- Deployment
- Code Organization

## COURSE CURRICULUM BY MODULE

MODULE	DESCRIPTION	WHAT YOU'LL LEARN
<i>Module 1: The Browser (Weeks 1-6)</i>	When most people think of the “Internet,” their mind immediately conjures up their web browser. We dive into detail about how the browser works and what exactly the source code that comprises a web page does.	<ul style="list-style-type: none"><li>• Creating a webpage from scratch</li><li>• Mastering terminal commands</li><li>• JavaScript and its most beloved child, jQuery</li></ul>
<i>Module 2: API Interaction (Weeks 7-10)</i>	The advent of the API has rapidly accelerated the pace of innovation in technology. Being able to communicate with other systems enables you to do even more with yours.	<ul style="list-style-type: none"><li>• Consuming RESTful API's</li><li>• Parsing JSON to extract meaningful data</li><li>• Using AJAX to update data on a website without having to hit the “refresh” button in the browser</li></ul>
<i>Module 3: The Joy of Serving (Weeks 11-16)</i>	Have you ever wondered where websites come from? They typically come from computer programs called “servers,” but did you know that servers do so much more? They interact with other databases and even other servers! Learn how to write server-side JavaScript code with node.js.	<ul style="list-style-type: none"><li>• Writing node.js server code to serve static web pages</li><li>• Scraping large amounts of data and storing into a MySQL Database</li><li>• Understanding relational databases</li></ul>
<i>Module 4: Full-Stack Development (Weeks 16-21)</i>	This is it. This is what you came for. Having your browser communicate with your server and your server with your database and back and forth via JSON and APIs—it's all making sense! If the pieces fit, you must code it!	<ul style="list-style-type: none"><li>• Going solo and creating something beautiful—user interface, server, database</li><li>• Working with teams to define responsibilities</li></ul>
<i>Module 5: Final Project (Weeks 22-24)</i>	Throughout the course, you've developed quite the respectable portfolio of projects to show future employers. This final project is all yours. It's as real as real can be—working in a group to create something truly magnificent from just an idea.	<ul style="list-style-type: none"><li>• Dreaming up something fantastic and understanding the bounds of reasonable and achievable</li></ul>

## CONTACT US

[codingbootcamp@docs.rutgers.edu](mailto:codingbootcamp@docs.rutgers.edu)  
(732) 430-2144