1. Welcome to the Full-Stack Engineer Path

   The first steps to earning yourself a full-stack engineering job.

2. Fundamentals of HTML

   Learn to build a structure for a website using HTML.

3. Fundamentals of CSS

   Learn how to apply styles to HTML documents using CSS.

4. Developing Websites Locally

   Get acquainted with the tools used to build websites outside the Codecademy IDE.

5. Deploying Websites

   Learn how to deploy websites using GitHub Pages and the command line.

6. Improved Styling with CSS

   Use intermediate-level CSS to customize styles and implement navigation elements.

7. Making a Website Responsive

   Learn how to utilize responsive web design practices using CSS grid, flexbox, and media queries.

8. JavaScript Syntax, Part I

   Use JavaScript to create basic programs that can store and manipulate various types of data.

9. JavaScript Syntax, Part II

   Explore JavaScript to create more complex programs using arrays, loops, objects, and iterators.

10. Git and GitHub, Part I

    Explore Git and GitHub to version control your programs.

11. Portfolio Project: JavaScript Syntax

    Build a Node.js console app that generates random messages each time a user runs the program and version your project with Git/GitHub.

12. Building Interactive Websites

    Implement JavaScript to add interactive experiences to a website.

13. Making A Website Accessible

    Get acquainted with the best accessibility (a11y) practices to ensure that your websites are usable to everyone.

14. Portfolio Project: Personal Portfolio Website

    Bring together what you have learned in the previous lessons and build a project off of Codecademy.

15. JavaScript Syntax, Part III

    Level up your JavaScript by learning how to implement classes, modules, and error handling.

16. JavaScript Testing

    Learn various JavaScript testing methodologies, such as Test-Driven Development (TDD), using Mocha and Chai.

17. Async JavaScript and HTTP Requests

    Learn about APIs (Application Programming Interfaces). Working with APIs will enable you to work with data stored on remote servers.

18. Web Apps

    Be introduced to web applications and learn about single page applications (SPAs) and how they are different from static websites.

19. React, Part I

    Be introduced to React, a popular JavaScript library, to build scalable front-ends for your website through the creation of components.

20. React, Part II

    Dive further into React and learn about props, state, hooks as well as testing with Jest and Enzyme.

21. Redux

    Learn how to manage React application's states with Redux.

22. Git and GitHub, Part II

    Learn how to collaborate on a project using Git and GitHub through forking, branching, merging conflicts, and making pull requests.

23. Portfolio Project: Reddit Client

    Bring together what you have learned in the previous lessons and build a project off of Codecademy.

24. Introduction to Back-End

    Start learning about back-end development and programming servers.

25. Build a Back-end with Express.js

    Learn the Node.js back-end environment, then create servers and APIs using Express.js

26. User Authentication & Authorization

    Learn how to protect your resources with authorization and authentication techniques.

27. Back-End and Feature Testing

    Learn about TDD techniques for full-stack web applications.

28. Database Basics

    Get acquainted with databases and how they help store, retrieve, and manipulate data.

29. Working with Databases

    Craft sophisticated SQL queries and calculations to build data-intensive applications.

30. Designing Relational Databases

    Learn how to design relational databases that you can then implement in PostgreSQL.

31. Advanced PostgreSQL

    Learn techniques for efficiently accessing data and maintaining optimal performance.

32. Adding a PostgreSQL Database

    Learn how databases store data so that it can be displayed on the web using PostgreSQL.

33. API Development with Swagger and OpenAPI

    Learn how to develop APIs using Swagger tooling and the OpenAPI specification.

34. Portfolio Project: E-Commerce App (REST API)

    Build an e-commerce REST API using Node/Express and PostgreSQL.

35. Connecting Front-End to Back-End

    Combine your front-end and back-end systems to create a cohesive full-stack application.

36. Deploying Web Applications

    Use Heroku to deploy a back-end application as well as a full-stack application.

37. Web Security Fundamentals

    Learn intermediate concepts in web security and prevention techniques for common threats.

38. Data Security

    Secure your data using Transport Layer Security, Role-Based Access Control, and more!

39. Common Attacks on Web Applications

    Explore common threats that web applications face and how to mitigate them.

40. Fundamentals of Operating Systems

    Learn about operating systems by taking a deep dive into each of its main functionalities.

41. DevOps Fundamentals

    Learn the fundamentals of DevOps, a culture supported by practices and tools.

42. Portfolio Project: E-Commerce App (Client)

    Expand the e-commerce REST API with a React client app, creating a PERN (Postgres, Express, React, Node) full-stack e-commerce experience.

43. What To Expect in a Technical Interview

    Know what to expect when interviewing for engineering jobs.

44. Linear Data Structures

    Begin interview preparation by learning the fundamentals of linear data structures.

45. Complex Data Structures

    Continue interview preparation by learning advanced applications of data structures.

46. Algorithms

    Continue interview preparation by learning the fundamentals of algorithms.

47. Search & Graph Search Algorithms

    Continue interview preparation by learning the fundamentals of search algorithms.

48. Interview Skills

    Review the key technical interview topics and soft skills that employers are looking for.

49. Portfolio Project: Open-Ended PERN App

    Build a PERN app of your choosing and deploy it to Heroku.

50. Next Steps in Your Full-Stack Engineer Journey

    Complete your Full-Stack Career Path.