High-Level Program Design

Software Engineering Immersive

GENERAL ASSEMBLY

Scale Engineering Capabilities

Software Engineering Immersive

Instructor-Led | Onsite or Remote | ~480 Hours

Overview:

Reskill non-technical talent into full-stack engineering roles to make immediate contributions to software projects. Participants will gain a solid base of fundamental programming, computer science knowledge, and experience with languages, frameworks, and libraries that the industry demands.

Business Outcomes:.

- Build junior engineering capacity within your tech team.
- Transform non-technical talent into job-ready software engineers.
- Solve real engineering problems through capstone projects.

~480 Hours of Software Engineering Training

Front-End Development	Full-Stack Development	Front-End Frameworks	APIs & Full-Stack Development	Capstone Project
Reinforce foundational pre-work topics such as HTML, CSS, JavaScript, and APIs.	**Build full-stack web applications** and deepen your knowledge of server-side development.	**Branch out into more advanced** JavaScript frameworks and concepts.	**Leverage third-party APIs** and practice executing a real-world workflow.	**Build and deploy a full-stack app** while mimicking a team-client interaction.

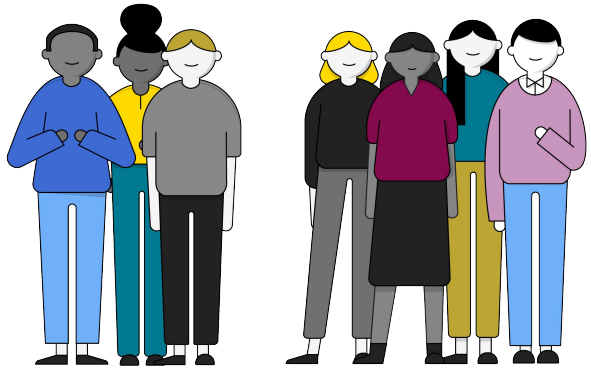
Why Software Engineering Immersive?

- **Validated tools and approaches** for modern, full-stack software engineering, developed in partnership with top organizations.
- **Built with subject matter experts** with experience in web application development, database management, cloud computing, and more.
- 480 hours of expert-led, hands-on learning, including:
    - **Projects, labs, and assignments** that mimic real-world tasks and workflows.
    - **Regular feedback and touchpoints** with instructors and peers to ensure students are meeting learning goals.

Learner Personas for Software Engineering Immersive

This product is specifically designed for the following audiences:

- **Undeclared Majors:** People who are looking for direction, structure, and employable skills at a lower cost than a 4-year college degree or traditional institution
- **Career Switchers**: Coming from a more creative or non-technical background and are often looking for technical skills they need to implement their designs.



Projects in Software Engineering Immersive

What to know about the **projects** for this course:

- Create a front-end application that powers engaging user interactions.
- Create a full-stack web application using JavaScript, including a database and user authentication.
- Create a full-stack app that leverages React for its front-end user interface.
- Learn a second programming language and create a full-stack application using a new, back-end framework in addition to React.

Anatomy of SEI

Unit	What's Covered	Unit Project
Unit 1: Front-End Web Development (4 weeks) Learn the foundational web technologies.	• Javascript programming syntax and basics • HTML and CSS • Object-oriented programming • DOM Manipulation • API calls	Create a JavaScript powered browser game that uses complex programming logic to drive an engaging user experience.
Unit 2: Building Applications With Express (3 weeks) Leverage the Express framework to create full-stack applications that include Mongo databases.	• Express • Mongo • Mongoose • bCrypt	Create a full-stack web application that allows users to create and manage resources stored in your mongo database.

Anatomy of SEI

Unit	What's Covered	Unit Project
Unit 3: React (2 weeks) Learn the most popular front-end framework.	• React set-up and syntax • React patterns, such as unidirectional data flow and components	Create a full-stack application that uses a backend API and relies entirely upon React to serve the user interface.
Unit 4: Second Language (2 weeks) Create an API using a second programming language and framework.	• Second language programming fundamentals • Opinionated web development framework, such as Django or Rails	Create a back-end API in your new programming language that connects to a dynamic, React front-end application.
Capstone Project (1 week)	• Tackle any new topics that present promising new areas of development post-course	Work with your instructors to design your own final project!

Tools Used in the Course

- Github
- A text editor, such as VS Code
- Node.js
- Mongo and SQL Databases
- Slack, Zoom, and other course communication tools

Software Engineering Immersive

What You'll Learn

Module Overview

Front-End Web Development

4 weeks 

Why?

HTML, CSS, and JavaScript are the foundational elements of every web experience. Through JavaScript, you'll learn software engineering fundamentals that are universal to any programming application.

Learning Objectives

- Create well-structured HTML pages
- Use CSS to apply dynamic style to web pages
- Use JavaScript to create interactive web pages with programming logic

Module Overview

Full-Stack JavaScript Development

3 weeks 

Why?

Once you start creating a full-stack application, with database storage and user sessions, you'll start to unlock the full potential of web development! You'll be able to create "real" apps that users can interact with in exciting new ways.

Learning Objectives

- Create web server applications using Express.js
- Use MongoDB for persistent data storage
- Use Mongoose to connect your Express application to MongoDB
- Manage users in an application, leveraging bCrypt for secure password storage

Module Overview

React

2 weeks 

Why?

React is the most popular front-end framework in web development today, and one of the most in-demand, job-relevant skills you'll learn in this course.

Learning Objectives

- Create React applications to serve as user interfaces for full-stack applications
- Understand React programming patterns, such as unidirectional data flow, component-based architecture, and functional programming concepts.

Module Overview

Second Programming Language

2 weeks 

Why?

Each software engineering team uses their own combination of tech tools, including different programming languages and frameworks. It's vital to be comfortable acquiring new languages and tools in order to adapt to any work environment.

Learning Objectives

- Understand the syntax and structure of a second programming language
- Understand the structure of SQL databases and create SQL queries
- Use an opinionated web development framework to create an API using a second programming language.

Module Overview

Capstone Project

1 week 

Why?

The end of this course is still just the beginning of your journey as a software engineer. Owning an individual project that stretches your self-guided learning skills will provide an effective off-ramp from SEI into the world of software engineering.

Learning Objectives

- Create a full-stack application that applies a new technology or techniques
-


GA