Software Development

Program Brochure · 2026

FULL STACK DEVELOPMENT

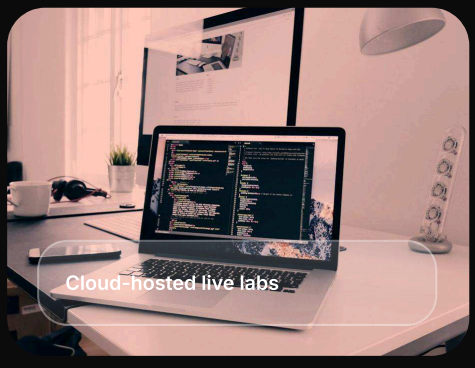
From idea to deployed product.

Advanced ↗ | 160 Hours ↗ | SAM-FSDEV ↗

Software Development

Code that ships.

18 programs in this track



8

modules · hands-on labs · capstone project

Inside this brochure

Contents

Program overview 03
Who it's for 04
Learning outcomes 05
Program runway 06
Detailed curriculum 07
Tools & frameworks 10
Projects & capstone 11
Certification & method 12
Learning path 13
FAQs 14



AT A GLANCE

Level	**Advanced**
Duration	**160 Hours**
Modules	**8**
Certificate	**SAM-FSDEV**

Program overview

From idea to deployed product.

Build and deploy complete web applications — frontend, APIs, database and deployment.

Teams increasingly value developers who can take a product from idea to deployment and own it end to end. This advanced program goes beyond stack basics into production engineering: TypeScript across React and Node, scalable API design, MongoDB performance, authentication, caching, testing, Docker, CI/CD, cloud deployment and observability, delivered through a substantial product build.

DELIVERY

ILT · VILT · Blended



WHAT YOU'LL PRACTISE

- Build React frontends
- Create Node and Express APIs
- Model data in MongoDB
- Deploy with Docker and CI/CD

LEVEL

Advanced

DURATION

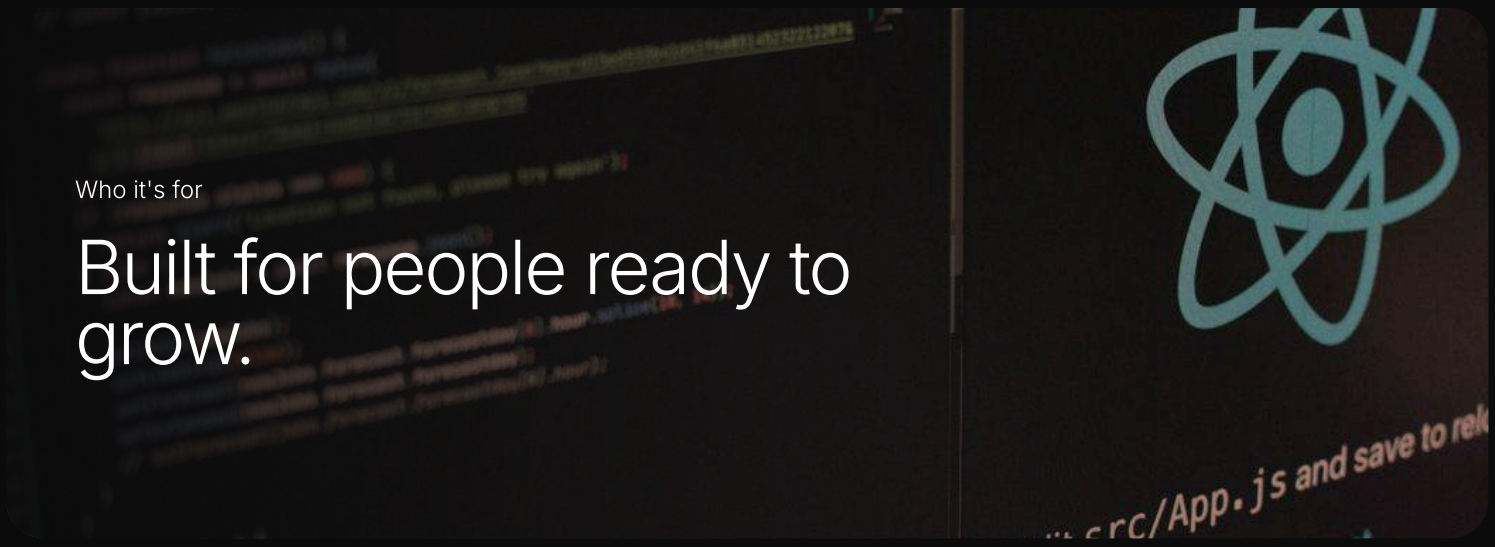
160 Hours

CERTIFICATION

SAM-FSDEV

TRACK

Software Development



Who it's for

Built for people ready to grow.

01

Students & freshers

Python, Java, C/C++ and JavaScript foundations

02

Frontend developers

React, Angular and TypeScript

03

Backend developers

Node, Django, FastAPI and Spring Boot

04

Senior engineers

Microservices, system design, leadership

PREREQUISITES

- Working knowledge of JavaScript, React or Node.js 01
- Prior experience building at least one web app 02
- Comfort with Git, npm and the command line 03

ROLES THIS PREPARES YOU FOR

- Full Stack Developer ↗
- Software Engineer (Web) ↗
- Product Engineer ↗
- Frontend or Backend Engineer ↗
- Technical Founder or Freelance Developer ↗

Learning outcomes

What you'll be able to do.



01	Architect full-stack applications using TypeScript across all layers
02	Build performant React frontends with Next.js and server-state caching
03	Design scalable Node.js APIs with validation and clean layering
04	Optimise MongoDB schemas, indexes and aggregation queries
05	Secure applications with OAuth, JWT and OWASP practices
06	Ship with Docker, CI/CD, cloud hosting and monitoring

Program runway

Module by module.

8
modules · 160 hours

01	**Engineering Foundations and TypeScript** Project architecture and monorepos, TypeScript for full-stack apps, Git branching and pull requests	16 hrs
02	**Advanced React and Next.js** React 19 hooks and patterns, Next.js App Router and layouts, Server and client components	28 hrs
03	**Node.js and Express API Engineering** Layered architecture: routes, services, repositories, Validation and error handling, REST design and OpenAPI	24 hrs
04	**MongoDB Design and Performance** Schema design patterns, Mongoose with TypeScript, Indexing and query explain plans	18 hrs
05	**Authentication, Security and Payments** JWT, refresh tokens and sessions, OAuth login with Auth.js, Role and permission models	18 hrs
06	**Performance, Caching and Testing** Redis caching strategies, Rate limiting and pagination, Unit tests with Vitest	18 hrs
07	**Docker, CI/CD and Cloud** Dockerfiles and multi-stage builds, Docker Compose for local stacks, GitHub Actions pipelines	20 hrs
08	**Capstone Product Studio** Agile sprint planning, Code reviews and pairing, Demo and documentation	18 hrs
★	**Capstone · Production SaaS Product Launch** Assessment, capstone review and certification	

The syllabus.

Modules 01–03

Module 01

Engineering Foundations and TypeScript

16 hours

Hands-on lab
Set up a TypeScript monorepo with shared types and lint rules.

TOPICS COVERED

- Project architecture and monorepos
- TypeScript for full-stack apps
- Git branching and pull requests
- Code quality with ESLint and Prettier
- Requirements to technical design

Git TypeScript

Module 02

Advanced React and Next.js

28 hours

Hands-on lab
Build the product's core frontend with Next.js App Router.

TOPICS COVERED

- React 19 hooks and patterns
- Next.js App Router and layouts
- Server and client components
- TanStack Query and Zustand
- Forms with React Hook Form and Zod
- Accessibility and responsive design

React

Module 03

Node.js and Express API Engineering

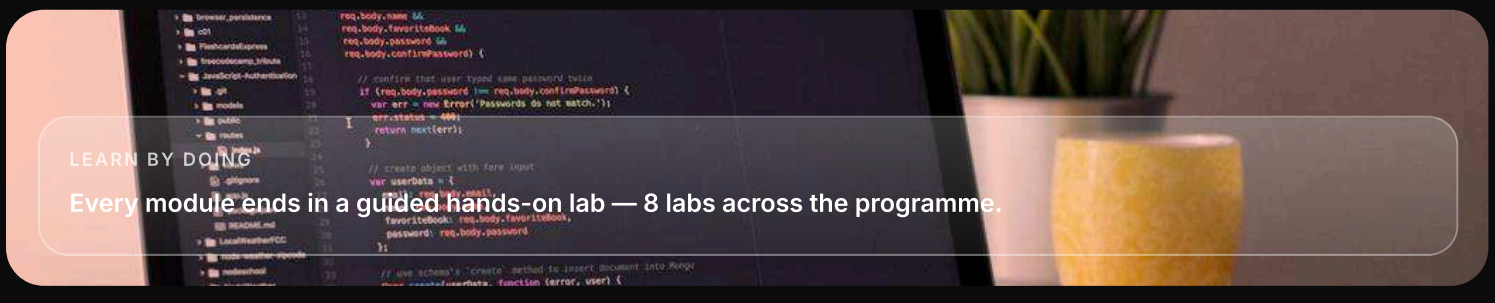
24 hours

Hands-on lab
Build a layered, documented Express API with background jobs.

TOPICS COVERED

- Layered architecture: routes, services, repositories
- Validation and error handling
- REST design and OpenAPI
- File uploads and object storage
- Background jobs with BullMQ
- Real-time with Socket.IO

Node.js Express



LEARN BY DOING

Every module ends in a guided hands-on lab — 8 labs across the programme.

Syllabus, continued.

Modules 04–06

Module 04

MongoDB Design and Performance

18 hours

Hands-on lab
Optimise slow queries using indexes and explain plan analysis.

TOPICS COVERED

- Schema design patterns
- Mongoose with TypeScript
- Indexing and query explain plans
- Aggregation pipelines
- Transactions and data consistency

MongoDB TypeScript

Module 05

Authentication, Security and Payments

18 hours

Hands-on lab
Add OAuth login, permissions and test-mode payment webhooks.

TOPICS COVERED

- JWT, refresh tokens and sessions
- OAuth login with Auth.js
- Role and permission models
- OWASP Top 10 for web apps
- Payment integration with webhooks

OWASP

Module 06

Performance, Caching and Testing

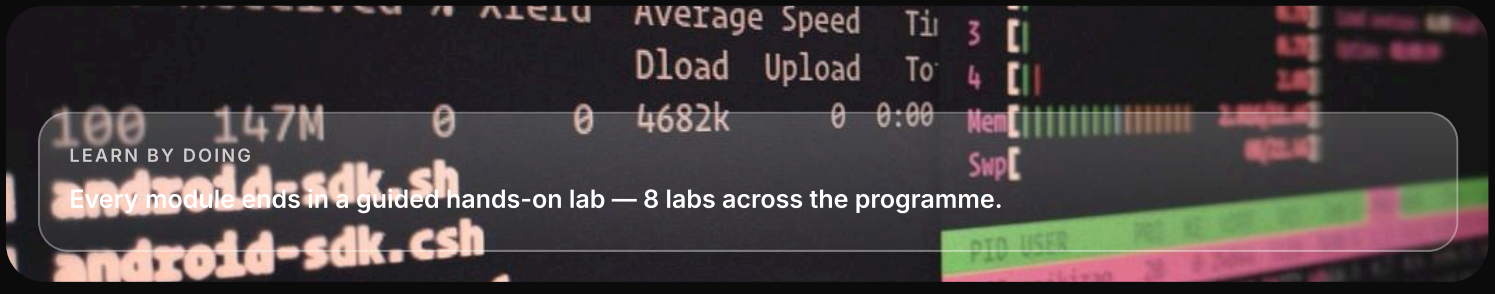
18 hours

Hands-on lab
Add Redis caching and a full automated test suite.

TOPICS COVERED

- Redis caching strategies
- Rate limiting and pagination
- Unit tests with Vitest
- API tests with Supertest
- End-to-end tests with Playwright
- Frontend performance and Core Web Vitals

Redis



Syllabus, continued.

Modules 07–08

Module 07

Docker, CI/CD and Cloud

20 hours

Hands-on lab
Build a CI/CD pipeline deploying the containerised stack to cloud.

TOPICS COVERED

- Dockerfiles and multi-stage builds
- Docker Compose for local stacks
- GitHub Actions pipelines
- Deploying to AWS, Render or VPS
- Nginx reverse proxy and HTTPS
- Logging, monitoring and error tracking

GitHub · Docker · AWS

Module 08

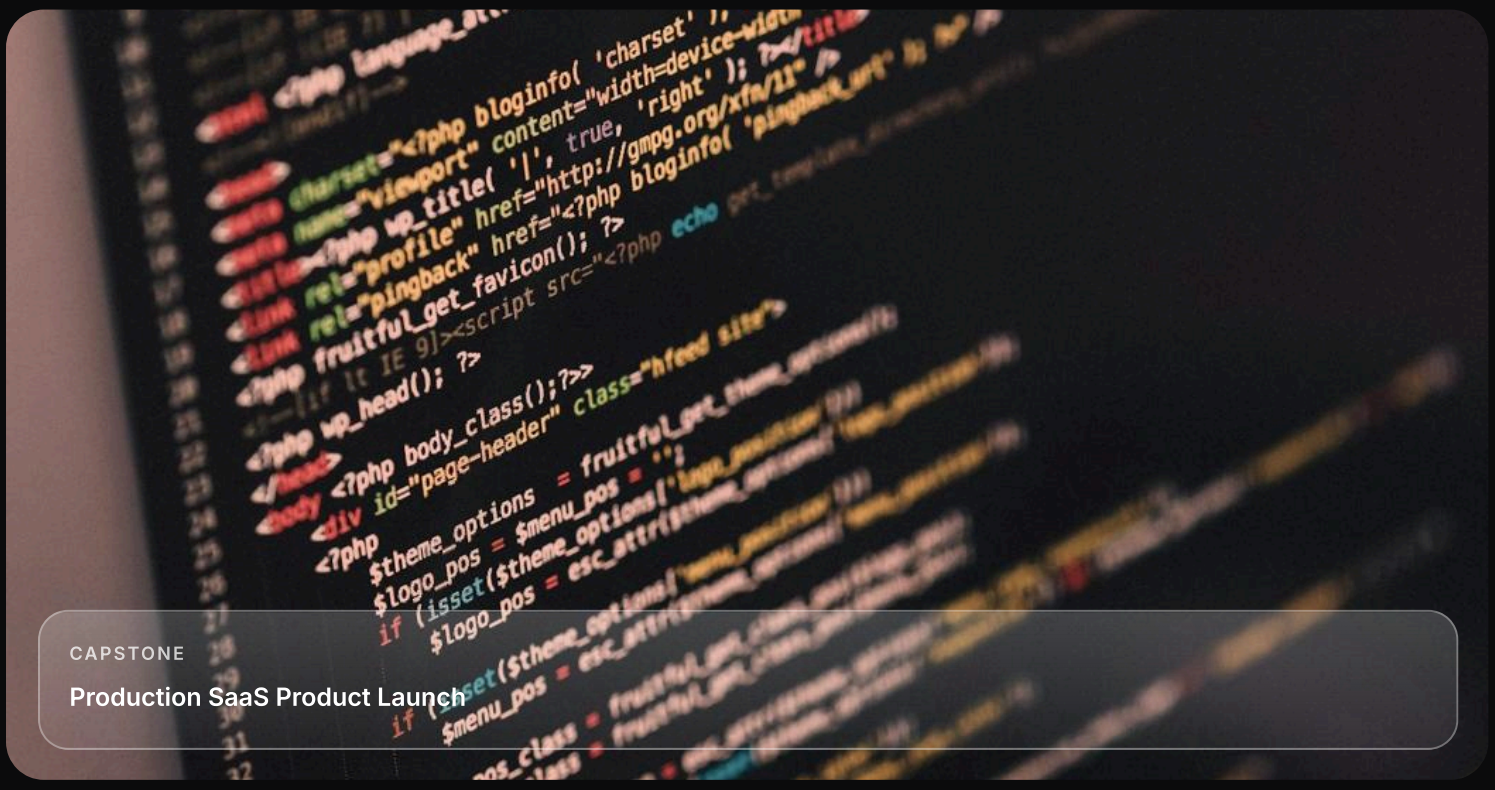
Capstone Product Studio

18 hours

Hands-on lab
Deliver and present your deployed capstone with technical documentation.

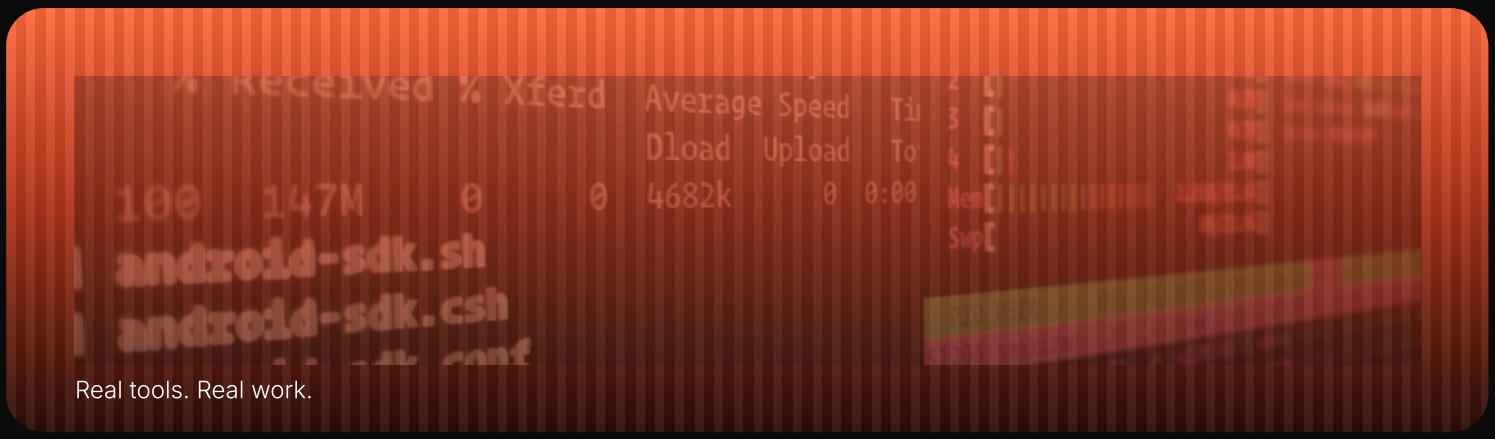
TOPICS COVERED

- Agile sprint planning
- Code reviews and pairing
- Demo and documentation
- Portfolio and case study writing
- Final assessment



CAPSTONE

Production SaaS Product Launch



Real tools. Real work.

React
Frontend UI library

Node.js
JavaScript runtime

MongoDB
Document database

Docker
Containers & images

Git
Version control

OWASP
Top 10 & testing standards

GitHub
Code hosting & Actions CI

AWS
Cloud platform

Redis
In-memory data store

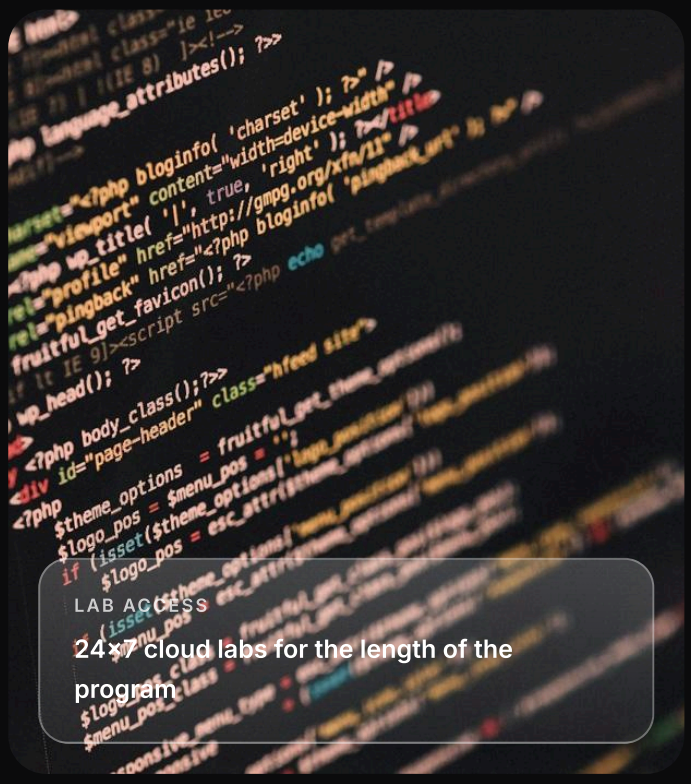
Express
Node web framework

Frameworks & standards

OWASP

FOCUS AREAS

Build React frontends	01
Create Node and Express APIs	02
Model data in MongoDB	03
Deploy with Docker and CI/CD	04



LAB ACCESS

24×7 cloud labs for the length of the program

01

SaaS Landing and Dashboard

A Next.js marketing site and authenticated dashboard with OAuth login, role-based views and server-side data fetching.

02

Real-Time Collaboration Board

A Trello-style board with drag-and-drop, live updates over Socket.IO, MongoDB persistence and optimistic UI updates.

03

Payments-Enabled Booking API

A booking backend with availability logic, test-mode payment webhooks, background email jobs and Redis caching.



Capstone project

Production SaaS Product Launch

Plan, build and deploy a multi-user SaaS application with a Next.js frontend, typed Node API, MongoDB, OAuth, payments, caching, automated tests, Dockerised CI/CD and monitoring, presented as a portfolio case study.

Real-world scenario ↗ Mentor review ↗ Portfolio ready ↗

Certification

SAM-FSDEV

Awarded on successful completion of the program assessments.

- MCQ assessments
- Practical lab evaluation
- Project & capstone review
- Final presentation



The SAM Learning Framework™

Scenario · Apply · Master · Mentor

Phase 01

Scenario

Real-world attack and use-case scenarios open every module.

Phase 02

Apply

Hands-on lab work, CTF challenges and live projects.

Phase 03

Master

Assessment, peer review and certification prep.

Phase 04

Mentor

Post-program career guidance and support.

DELIVERY MODELS

- Instructor-Led (ILT)
- Virtual Instructor-Led
- Self-Paced Online
- E-Learning Licensing
- Blended Learning
- Corporate Onsite
- Bootcamp Intensive

Learning path · Software Development

Where this program fits.

Beginner	Intermediate	Advanced	Expert
4 programs	10 programs	You are here	1 programs

Before this

Python Advanced (OOP, Design Patterns)	56 Hours
TypeScript Mastery	40 Hours
Django Web Framework	56 Hours
FastAPI — Modern Python APIs	48 Hours

Up next

Software Engineering Leadership	48 Hours



Questions, answered.

Q01

Do I need prior experience to join?

Yes. This advanced program expects working JavaScript skills and some prior experience with React or Node.js, ideally having built at least one small web application. Fundamentals are revised quickly, not taught from scratch.

Q02

How are classes scheduled and delivered?

Sessions are instructor-led and run as live virtual or blended batches, with weekday and weekend options where available. Each session combines concept walkthroughs, live coding and guided lab time, with recordings or notes shared for revision.

Q03

Will I receive a certificate?

Yes. Learners who complete the required labs, projects and final assessments receive the Samcommunity Academy SAM-FSDEV certificate. It is awarded on the basis of assessed work, not attendance alone.

Q04

What tools and lab access will I need?

You need a laptop with at least 8GB RAM, Node.js LTS, VS Code, Git and Docker Desktop, plus free accounts for MongoDB Atlas, GitHub and a cloud hosting tier.

Q05

Is career support included?

You get guidance on building a portfolio from your projects, resume and GitHub profile reviews, and mock technical interviews. We help you prepare and present your skills, but we do not guarantee jobs, placements or salary outcomes.



Learn practical skills. Grow with mentors.

Xylyn Samcommunity Private Limited · DPIIT-recognized startup · Surat, Gujarat, India