SIFT GLOBAL
SCHOOL OF INNOVATION AND FUTURE TECHNOLOGIES
LEARN · INNOVATE · LEAD

DISTANCE LEARNING PROGRAMME

NEW INTAKE NOW OPEN · LIMITED SEATS

Never written code? Become a Full Stack Developer.

The **Advanced Certificate in Full Stack Development** takes you from zero coding knowledge to building and launching real web applications — then puts you to work on real projects in a **3-month industry internship**.



Zero coding experience needed	**24** live classes · 3 months	**1 real app** built & launched online	**3 months** internship training

HTML & CSS · JavaScript · React · Java · Spring Boot · MySQL · Git & GitHub · Security (JWT) · Cloud deployment · Career skills

Enrol now · Rs. 29,000
Special offer — usual price ~~Rs. 35,000~~ · call or WhatsApp to reserve

HOTLINE 071 383 8638
WHATSAPP 071 383 8637

TECHNOLOGY PARTNERS

MERN CREST SOLUTIONS

WebFix

MernCode Academy

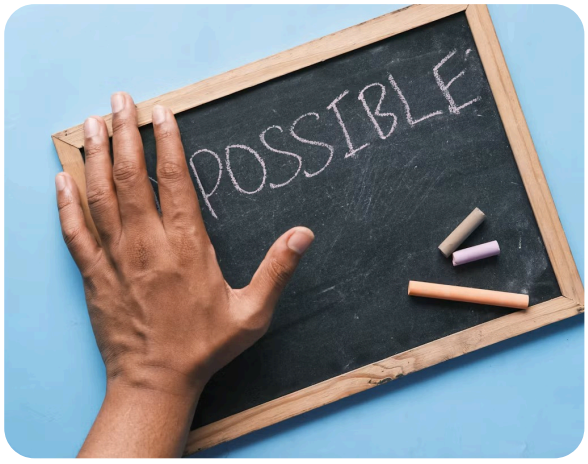
MADE FOR COMPLETE BEGINNERS

No coding knowledge? We start from zero.

You do **not** need to know anything about code, computer science or maths to join. If you can use a computer, type and browse the internet, you can learn this.

We start with the very first step — *what is a website and how does it work?* — and build up one small step at a time. Every new idea is explained in simple language, shown live on screen, and then you try it yourself with your lecturer's help.

By the end you will not just "know about" coding. You will have **built a complete web application with your own hands**, put it online, and used it with real users — the same work professional developers do every day.



Three simple phases

01
Learn

24 LIVE CLASSES · 3 MONTHS

Over 3 months (8 classes a month), two lecturers teach you the front of a website (what users see) and the back (the logic and data), step by step from zero.

02
Build

CAPSTONE PROJECT

Every session adds a feature to one real app. You finish with a working, secure task-management app published online.

03
Work

3-MONTH INTERNSHIP

Join a real software team, work on real projects with a mentor, and earn genuine workplace experience for your CV.

Who is this course for?

- **Students after O/L or A/L** who want a career in IT
- **University students** who want real project experience
- **Working adults** who want to move into the tech industry
- **Business owners** who want to build their own systems
- **Freelancers** who want to offer web development services
- **Anyone curious** who has always wanted to learn to code

> **Our promise to beginners:** nobody is left behind. Every class is recorded, every lab has step-by-step guidance, and mentors are available to help you fix errors when you get stuck — getting stuck is a normal part of learning to code.

EXPLAINED IN SIMPLE WORDS

What is “Full Stack” development?

Think of a web application like a **restaurant**. A Full Stack Developer can build **every part** of the restaurant — not just one.

FRONTEND
The dining area & menu
Everything the customer sees and touches. Built with **HTML, CSS, JavaScript & React**.

→

API
The waiter
Carries orders to the kitchen and brings food back. Called a **REST API**.

→

BACKEND
The kitchen
Where the real work happens: rules, checks and security. Built with **Java & Spring Boot**.

→

DATABASE
The storeroom
Where everything is kept safely and found quickly. We use **MySQL**.

Most developers learn only one part. In this course you learn **all of them** and how they connect — which is why Full Stack Developers are valued by employers and can build complete products on their own.

Words you will hear in class

Code — written instructions that tell a computer what to do.

HTML — the structure of a web page (headings, text, images, buttons).

CSS — the design: colours, fonts, layout, mobile-friendly screens.

JavaScript — makes pages interactive: clicks, forms, live updates.

React — a popular tool for building modern app screens quickly.

Java — a powerful language used by banks and large companies.

Spring Boot — a toolkit that makes building Java backends fast.

MySQL / SQL — a database and the language to save and find data.

API — the messenger between the frontend and the backend.

JSON — the simple text format apps use to send data.

Git & GitHub — saves every version of your code and shows it to employers.

Deploy — putting your app on the internet so anyone can use it.

Bug / Debugging — a mistake in code / finding and fixing it.

Login & JWT — how apps check who you are and keep data private.

Don't worry if these words are new — by the end of the course you will use every one of them confidently.



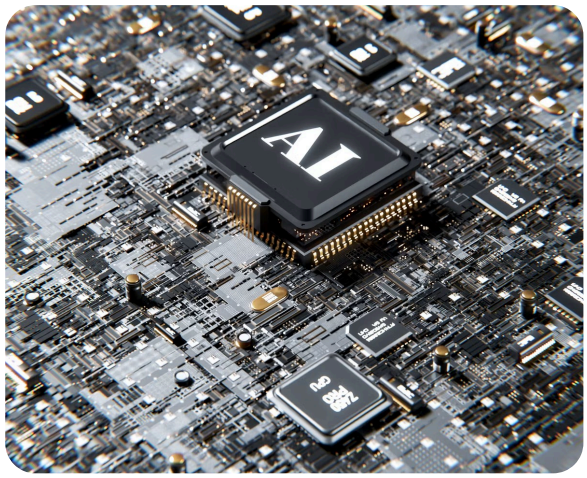
LEARNING TO CODE IN THE AGE OF AI

AI can't build it for you. Skilled people can build anything with AI.

AI tools like ChatGPT, Copilot and Claude can write pieces of code in seconds. But a piece of code is not a working product.

To build a real application with AI, someone has to know **what to ask for**, **how the parts fit together**, whether the code AI gives back is **correct and safe**, and **how to fix it** when something breaks. Without that knowledge, AI-generated code quickly turns into errors you can't understand and security holes you can't see.

That knowledge is exactly what this programme gives you. **AI becomes your assistant — you stay the builder.**



AI on its own vs. you with skills + AI

AI tools on their own	You — with real skills — using AI
Write small pieces of code when asked	**Plan the whole application** and how every part connects
Don't know if the code suits your project	**Check, test and correct** AI code before it goes live
Can produce code with hidden security holes	**Protect passwords, data and users** properly
Get stuck in loops when errors appear	**Read the error, debug and fix it** yourself
Can't run, maintain or take responsibility for a live system	**Deploy, maintain and improve** real systems for real users

The four kinds of knowledge you need to build anything with AI

Technology
HTML, CSS, JavaScript, React, Java, Spring Boot and SQL — the languages AI writes in. You must read and understand them.

Stack
How the frontend, API, backend and database connect into one working system. AI can't see your whole system — you can.

IT fundamentals
How the internet, servers, cloud hosting, security and Git work — what keeps an app online and safe.

Skills
Problem solving, debugging, testing and teamwork — the abilities that turn code into a product.

Our approach — fundamentals first, AI as your power tool. In every session you first learn to build each part yourself. Then you learn to use AI assistants responsibly to work faster: writing better prompts, reviewing AI code, testing it and explaining every line. The result is a developer who can build anything — with or without AI.

YOUR JOURNEY

From day one to your first developer role

BEFORE CLASS STARTS

Get ready — Module 0

Set up your laptop with free tools and watch short beginner lessons. We guide you click by click.

MONTH 1 · SESSIONS 1–8

Foundations: web pages, Java & databases

Build and style real web pages, learn JavaScript and Java, design a database and start your Spring Boot server.

MONTH 2 · SESSIONS 9–16

Build the app: APIs & React

Make pages interactive, build REST APIs, save data with JPA, and create your app screens in React.

MONTH 3 · SESSIONS 17–24

Connect, secure, test & launch

Join frontend and backend, add login and security, write tests, deploy to the cloud and present your capstone.

END OF COURSE

Capstone Showcase & Certificate

Present your live app and receive the Advanced Certificate in Full Stack Development.

NEXT 3 MONTHS

Internship training

Work in a real software team on real projects, guided by a mentor.

THEN

Start your career

Apply for junior developer roles with a certificate, a live app, a GitHub portfolio and real experience.



Each week looks like this

- **2 live online classes** (2–3 hours each) — 8 classes a month
- **Practice lab** after each class with step-by-step instructions
- **Recordings** of every class to watch again
- **Mentor help** to fix errors and answer questions
- **4–6 hours** of your own practice per week

Learn from home. All classes are live on Zoom, so you can join from anywhere in Sri Lanka or abroad. Missed a class? Watch the recording.

THE SYLLABUS, EXPLAINED · MONTH 1 · SESSIONS 1–8

Foundations: web pages, Java & databases

Module 0 — Get Ready (before class, self-paced): we help you set up your laptop like a real developer's — VS Code, IntelliJ IDEA, JDK, Node.js, MySQL, Postman, Git — and teach the absolute basics: what a website is and how the internet works.

01 FRONTEND

Building Web Pages with HTML

In simple words: HTML is the skeleton of every website. You learn to place headings, text, images, links and forms on a page — and make them usable for everyone.

- ✓ Build a real sign-up form that checks what users type
- ✓ Save your work to GitHub

HTML5 semantic elements · forms · validation · accessibility · Git

02 BACKEND

Java Basics — Thinking Like a Programmer

In simple words: You write your first real programs — teaching the computer to store information, make decisions and repeat tasks.

- ✓ Write and run your first Java programs
- ✓ Use variables, conditions, loops and methods

Syntax · data types · operators · if/else · loops · methods

03 FRONTEND

Modern CSS Layouts

In simple words: CSS is the paint and furniture. You learn colours, fonts, spacing and the two layout systems every modern website uses.

- ✓ Design a professional product-card page
- ✓ Lay out pages with Flexbox and Grid

Selectors · box model · Flexbox · CSS Grid · typography

04 BACKEND

Object-Oriented Java

In simple words: Real programs organise information into "objects" — like a Task with a title and a due date. You learn the way professional Java code is structured.

- ✓ Create your own Task class and objects
- ✓ Manage lists of tasks with collections

Classes · constructors · inheritance · interfaces · List & Map

05 FRONTEND

Responsive Design & Tailwind CSS

In simple words: Your pages must look great on phones, tablets and laptops. You learn mobile-first design and Tailwind CSS, a fast modern styling tool.

- ✓ Make any page mobile-friendly
- ✓ Style pages quickly with Tailwind CSS

Media queries · mobile-first · Tailwind CSS utilities

06 BACKEND

Databases, SQL & MySQL

In simple words: A database is where an app keeps its information safely. You design tables and learn SQL, the language for saving and finding data.

- ✓ Design tables for users and tasks
- ✓ Write SQL queries to save and search data

Tables · primary & foreign keys · SELECT/INSERT/UPDATE/DELETE · joins

07 FRONTEND

JavaScript Fundamentals

In simple words: JavaScript is the brain of a web page. You learn the language that makes pages react — calculations, lists and decisions.

- ✓ Build a live product filter by price and category
- ✓ Find and fix bugs with browser tools

Variables · functions · arrays · objects · JSON · DevTools

08 BACKEND

Your First Server with Spring Boot

In simple words: A server waits for requests and answers them. Spring Boot lets you create one in minutes — the start of your app's "kitchen".

- ✓ Create the backend project for your capstone app
- ✓ Make your server reply to the browser

Spring Initializr · Maven · project structure · @RestController

THE SYLLABUS, EXPLAINED · MONTH 2 · SESSIONS 9–16

Building the app: APIs, JavaScript & React

09 FRONTEND

DOM, Events & Modern JavaScript

In simple words: The DOM is how JavaScript changes what you see on screen. You make buttons, forms and lists come alive when users click and type.

- ✓ Add, edit and remove items on the page
- ✓ Handle clicks, typing and form submits

DOM · events · ES6+ · modules · localStorage

10 BACKEND

Creating REST APIs

In simple words: An API is the waiter between screen and kitchen. You build the "menu" of requests your app accepts — add, view and delete tasks — and test them.

- ✓ Build working task APIs and test them in Postman
- ✓ Create automatic API documentation

REST · HTTP verbs · status codes · JSON · Postman · Swagger

11 FRONTEND

Talking to the Internet — Async JavaScript

In simple words: Real apps load live data — weather, products, messages. You learn to request data from the internet and show it with "loading..." and error messages.

- ✓ Show live data from a real online API
- ✓ Handle slow internet and errors gracefully

Promises · async/await · try/catch · fetch() · Axios

12 BACKEND

Saving Data with Spring Data JPA

In simple words: You connect your Java server to MySQL so tasks and users are saved permanently and linked together — without writing SQL by hand.

- ✓ Save and search data from Java automatically
- ✓ Link users to their tasks

Spring Data JPA · Hibernate · @Entity · relationships · repositories

13 FRONTEND

Introduction to React

In simple words: React lets you build screens from reusable "LEGO blocks" called components — the way Facebook, Netflix and Airbnb build their apps.

- ✓ Create a React app and your own components
- ✓ Display a list of tasks from data

Vite · JSX · components · props · lists & keys

14 BACKEND

Building Every Feature Properly — CRUD

In simple words: Professional apps are organised in layers, like departments in a company. You complete create, view, update and delete — with pages, sorting and search.

- ✓ Finish all task features in the backend
- ✓ Show 10 tasks per page, sorted and searchable

Controller → Service → Repository · CRUD · pagination & search

15 FRONTEND

Interactive Screens & Forms in React

In simple words: "State" is what the app remembers right now — what's typed, what's ticked. You build forms that add and edit tasks and update the screen instantly.

- ✓ Build add / edit task pop-up forms
- ✓ Check user input before saving

useState · controlled inputs · validation · custom hooks

16 BACKEND

Checking Data & Handling Errors

In simple words: Users make mistakes — empty titles, wrong dates. You teach your backend to check every request and reply with a clear message instead of crashing.

- ✓ Reject bad data with helpful error messages
- ✓ Keep a log of what happens in your app

DTOs · Bean Validation · @ControllerAdvice · logging

THE SYLLABUS, EXPLAINED · MONTH 3 · SESSIONS 17–24

Connect, secure, test & launch

17 FRONTEND
Connecting Frontend to Backend

In simple words: The big moment — your React screens start talking to your own Java server. Tasks you add on screen are saved in your database for real.

- ✓ Make your full app work end-to-end
- ✓ Show loading spinners and error notices

useEffect · GET/POST/PUT/DELETE · loading & errors · CORS

18 BACKEND
Login & Security

In simple words: Every real app needs a secure login. You protect passwords and make sure each user sees only their own tasks — just like banking apps.

- ✓ Build secure register and login
- ✓ Protect private data from other users

Spring Security · BCrypt · JWT tokens · roles · CORS

19 FRONTEND
Pages, Navigation & Login Screens

In simple words: You add multiple pages — Login, Dashboard, Settings — and lock private pages behind login, just like professional web apps.

- ✓ Add navigation between pages
- ✓ Protect private pages behind login

React Router · layouts · protected routes · auth context

20 BACKEND
Testing Your Code

In simple words: Professionals test their code before customers see it. You write automatic tests that catch mistakes before they reach users.

- ✓ Write automatic tests for your services
- ✓ Test your APIs end-to-end

JUnit 5 · Mockito · integration test concepts

21 FRONTEND
Publishing Your Frontend Online

In simple words: You prepare your React app for real users — a fast production build — and publish it on the internet with its own web link.

- ✓ Publish your app online for anyone to visit
- ✓ Use environment settings safely

.env · production build · Vercel / Netlify

22 BACKEND
Docker & Cloud Deployment

In simple words: You package your server and launch it, with its database, on a cloud server — so your full application is live on the internet.

- ✓ Package your backend as a container
- ✓ Deploy your API and database to the cloud

Spring profiles · secrets · JAR · Docker basics · cloud hosting

23 CAPSTONE
Integration & AI-Assisted Development

In simple words: You polish the complete app and learn to use AI coding assistants responsibly — writing better prompts, reviewing AI code and testing it.

- ✓ Use AI tools to speed up your work safely
- ✓ Review, fix and improve the full app

AI prompting · code review · refactoring · final testing

24 CAPSTONE
Capstone Showcase & Viva

In simple words: Your graduation moment. You present your live app to the lecturers, explain how you built it and answer questions — just like a job interview demo.

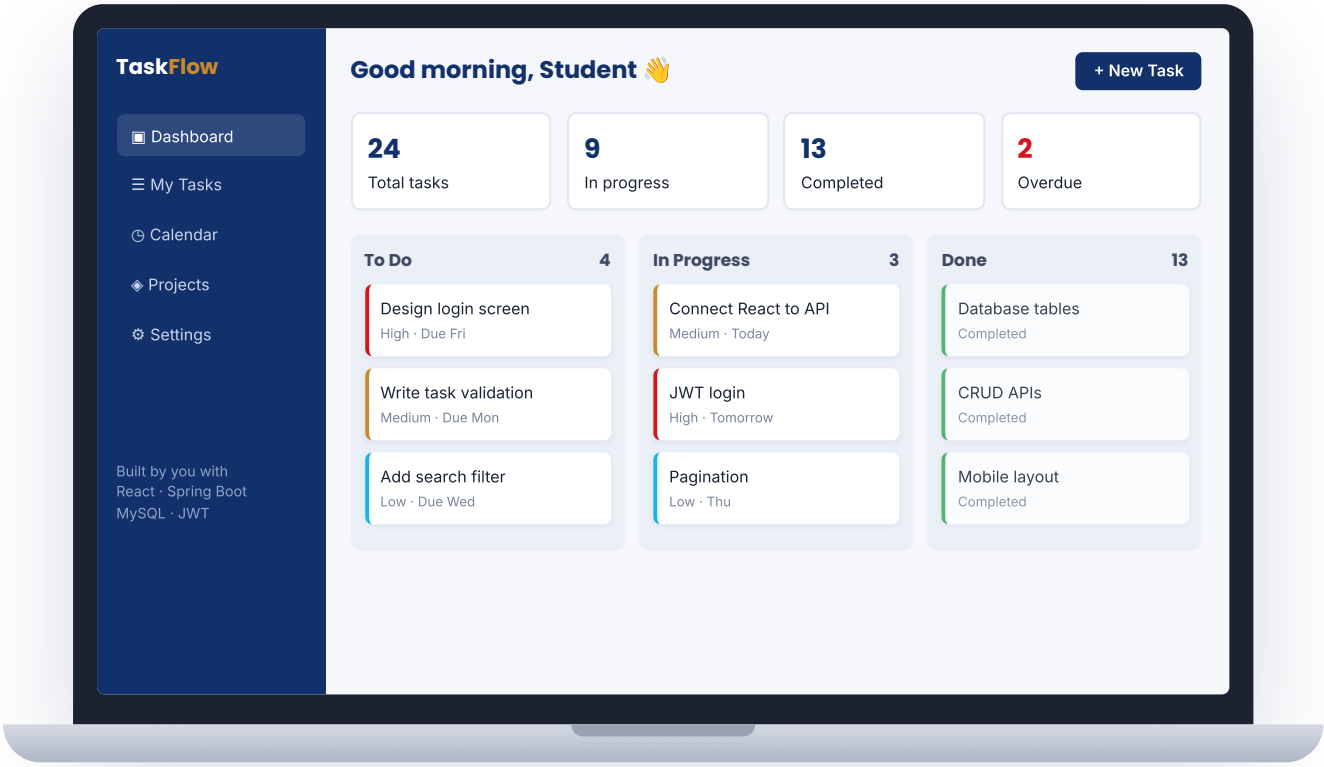
- ✓ Present a complete, working, online application
- ✓ Earn your Advanced Certificate

Live demo · code walkthrough · viva · portfolio review

YOUR CAPSTONE PROJECT

The real app you will build — TaskFlow

A complete task-management web app, like the tools used by companies to organise work. You build every part of it yourself and finish with a live link to share with employers.



Screens

Login, register, dashboard, add / edit forms — on mobile and desktop.

Brain

A Java backend that checks every request and keeps users' data separate.

Memory

A MySQL database that stores users and tasks permanently.

Live online

Deployed to the internet with a link you can put on your CV.

Why one real project? Employers don't ask “did you attend a course?” — they ask “show me what you built.” Your TaskFlow app, its code on GitHub and a short demo video become the centrepiece of your portfolio.

REAL-WORLD EXPERIENCE

3-Month Internship Training

Learning in class is step one. **Working on real projects** is what turns you into a developer. After your certificate, eligible graduates join three months of supervised internship training with our technology partners.

You'll work like a junior developer in a real team — taking tasks from a project board, writing code, getting it reviewed by senior developers, and delivering features that real users rely on.



HOST PARTNERS

MERNCREST SOLUTIONS

WebFix

M1

Learn the team

- Company tools & coding rules
- Set up the real project
- Fix small bugs with a mentor
- Daily stand-up meetings

M2

Build features

- Build React screens & Java APIs
- Work from a sprint board
- Code reviews with seniors
- Mid-point feedback

M3

Own & deliver

- Own a feature start to finish
- Test and help release it
- Write handover documents
- Final presentation

What you gain

- ✓ Internship completion certificate
- ✓ 3 months of real experience for your CV
- ✓ Mentor feedback and a reference for strong performers
- ✓ Contacts in the software industry

Eligibility

- ✓ Awarded the Advanced Certificate
- ✓ Capstone app completed and live
- ✓ Good attendance and conduct
- ✓ Short selection interview

Please note: this is an **unpaid** internship training placement — there is no salary or allowance. Placements depend on partner project availability, and the internship does not guarantee employment. Start dates, mode (online / hybrid) and weekly hours are confirmed in your offer letter.

YOUR CAREER

Job skills you'll walk away with

After the course and internship you'll have what entry-level developer job adverts ask for: **skills, proof and experience.**

- **A certificate** — Advanced Certificate in Full Stack Development
- **A live app** — your TaskFlow project online with a web link
- **A GitHub portfolio** — clean code employers can review
- **Real experience** — 3 months in a software team
- **A developer CV & LinkedIn** — written with our guidance
- **Interview practice** — a mock interview with feedback



Jobs you can apply for

Role	What you'd do	Skills from this course
Junior Full Stack Developer	Build features from screen to database	React, Spring Boot, MySQL, Git
Junior React / Frontend Developer	Build modern, mobile-friendly app screens	HTML, CSS, JavaScript, React
Junior Java Backend Developer	Build APIs, business logic and data storage	Java, Spring Boot, JPA, Security
Associate Software Engineer	Work in an Agile product team	Full stack, Git workflow, teamwork
QA / API Tester (trainee)	Test apps and APIs, report bugs	Postman, Swagger, SQL, JUnit
Freelance Web Developer	Build websites and systems for clients	Complete build + deployment

Workplace skills

Problem solving · Debugging · Agile / Scrum · Teamwork · Code reviews · Time management · Presenting your work · Using AI tools responsibly · Self-learning · Professional communication



WHY SIFT GLOBAL

Why students choose this programme

🎯 **Built for beginners**
Starts from zero with simple explanations, recordings and step-by-step labs.

🏢 **Industry partners**
Curriculum and internships with working software companies.

👩‍💻 **Two specialist lecturers**
A dedicated Frontend lecturer and a dedicated Backend lecturer.

🌍 **Study from anywhere**
Live online classes on Zoom with recordings on MernCode.online.

🛠️ **Learn by building**
Every session adds a real feature to your own app.

🚀 **Career support**
CV, LinkedIn, GitHub portfolio and mock interview included.

Programme details

Award	Advanced Certificate in Full Stack Development — SIFT Global
Mode	Distance Learning Programme — live online + recordings
Course length	3 months · 24 live classes (8 per month, 2–3 hrs each)
Course fee	**Rs. 29,000.00** ~~Rs. 35,000.00~~ special offer
Internship	3 months, unpaid training (eligible graduates)
Language	English, with support in Tamil & Sinhala
Entry	O/L recommended · no coding experience needed
Assessment	Labs 30% · Checkpoint 15% · Capstone 40% · Viva 15%
To pass	50% overall, 80% attendance, live capstone app

What you need

- A laptop or desktop (8 GB RAM minimum, 16 GB recommended)
- Stable internet for Zoom classes
- Microphone or headset (webcam recommended)
- 4–6 hours per week for practice

All software used in the course is **free**. Phones and tablets are not suitable for coding.

Frequently asked questions

I'm not good at maths. Can I still learn?
Yes. Web development needs logical thinking, which we teach step by step — not advanced maths.

Is the internship paid?
No. It is unpaid training that gives you real experience and a certificate.

What if I miss a class?
Watch the recording within 48 hours and complete the lab.

Do I need to buy software?
No. Every tool we use is free.

Can I study while working or at university?
Yes — classes suit busy students and every session is recorded.

Will I get a job?
We prepare you fully with skills, portfolio, CV and interview practice; jobs are not guaranteed.

LEARN FROM QUALIFIED LECTURERS

Meet your lecturers

Two dedicated lecturers — one for Frontend, one for Backend — teach you in alternating live sessions, so every topic is taught by a specialist.

TK

FRONTEND LECTURER

K. P. Tashini Kawindi

- **BSc (Hons) Computing**
- Coventry University, UK

Teaches the Frontend Track — 11 sessions
HTML5 · CSS layouts · Responsive design & Tailwind CSS · JavaScript · DOM & async JavaScript · React · React forms & state · Connecting to APIs · Routing & login screens · Frontend deployment

Sessions 1 · 3 · 5 · 7 · 9 · 11 · 13 · 15 · 17 · 19 · 21

MP

BACKEND LECTURER

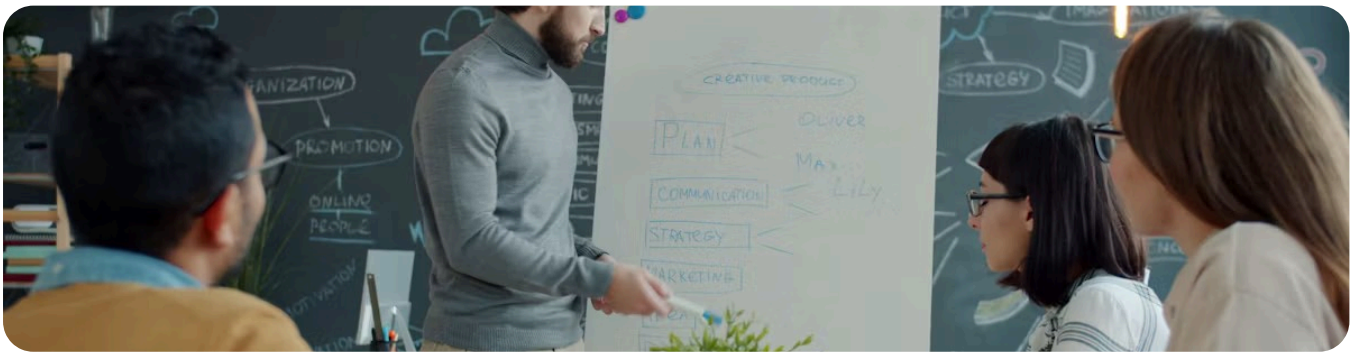
M.B.H.M.U. Pehesara

- **HND in Software Engineering**
- **BSc (Hons) Computing**
- Coventry University, UK

Teaches the Backend Track — 11 sessions
Java basics & OOP · SQL & MySQL · Spring Boot · REST APIs · Spring Data JPA · CRUD architecture · Validation & error handling · Spring Security & JWT · Testing · Docker & cloud deployment

Sessions 2 · 4 · 6 · 8 · 10 · 12 · 14 · 16 · 18 · 20 · 22

Capstone Sessions 23–24 are run jointly by both lecturers: integration, AI-assisted development and your final Capstone Showcase & Viva.



HOW TO ENROL

Course fee, bank details & receipt upload

COURSE FEE · SPECIAL OFFER

~~Rs. 35,000.00~~

Rs. 29,000.00

You save Rs. 6,000

Bank details for payment

Account name	**Mohammed Shakkir MSMS**
Bank	**Commercial Bank of Sri Lanka**
Branch	**Vavuniya Branch**
Account number	**8014425901**
Amount	**Rs. 29,000.00**
Reference	Your full name + mobile number

Enrol in 4 easy steps

1. **Pay the course fee** to the bank account by deposit, online transfer or CEFT.
2. **Fill in the confirmation form** below with your details.
3. **Upload your receipt** — send a clear photo or screenshot with your details to WhatsApp **071 383 8637**.
4. **Get confirmed** — receive your enrolment confirmation, class group link and learning platform login.

Payment confirmation & receipt upload

STUDENT FULL NAME

NIC NUMBER

MOBILE / WHATSAPP

EMAIL ADDRESS

AMOUNT PAID (RS.)

PAYMENT DATE

BANK / BRANCH PAID AT

SLIP / TRANSACTION REFERENCE NO.

Upload your payment receipt

Send a clear photo or screenshot of your bank slip / transfer receipt, together with the details above, to:

WhatsApp 071 383 8637

Printed copy? Attach the receipt in this box and hand it in with this form.

Your seat is reserved once your payment is verified. Please keep your original receipt until you receive confirmation. For help, call the hotline 071 383 8638.

SIFT GLOBAL

EST. 2026

SCHOOL OF INNOVATION AND FUTURE TECHNOLOGIES

LEARN • INNOVATE • LEAD

Your developer career starts with one decision.

Join the next intake of the Advanced Certificate in Full Stack Development — no coding experience needed. Seats are limited to keep classes small and supported.



COURSE FEE · SPECIAL OFFER

~~Rs. 35,000.00~~

Rs. 29,000.00

You save Rs. 6,000

Pay to **Commercial Bank · Vavuniya Branch · A/C 8014425901 · Mohammed Shakkir MSMS** — send your receipt to WhatsApp **071 383 8637**

YOUR LECTURERS

TK

K. P. Tashini Kawindi

Frontend Lecturer

BSc (Hons) Computing — Coventry University, UK

MP

M.B.H.M.U. Pehesara

Backend Lecturer

HND in Software Engineering · BSc (Hons) Computing — Coventry University, UK

HOTLINE

071 383 8638

WHATSAPP

071 383 8637

LEARN ONLINE

MernCode.online

MERN CREST SOLUTIONS

WebFix

Information correct at the time of publication (November 2026). Course content and schedules may be updated to stay aligned with industry needs. · Powered by MERNcrest Solutions (Pvt) Ltd — merncrest.lk

PART 2

Student Handbook

Important information every student must read before the first session — how the programme runs, the full technical curriculum, assessment rules, the internship and student policies.



Contents	Page
Welcome & programme at a glance	17
Entry requirements & equipment	18
How the Distance Learning Programme works	19
Technology stack	20
Course structure & Module 0 — Readiness	21
Detailed curriculum — Sessions 1–24 (3 months)	22
Capstone project & marking criteria	25
Professional skills, assessment & certification	26
3-month internship training	27
Job-ready skills & career pathways	29
Rules, academic integrity & FAQs	30
Before Session 1 — checklist & contacts	31

WELCOME

Welcome to SIFT Global

Thank you for choosing the **Advanced Certificate in Full Stack Development**, delivered through the SIFT Global Distance Learning Programme (DLP).

This handbook explains everything you need to know **before** your first session: how the course is delivered, what you will learn each week, how you will be assessed, what the 3-month internship training involves, and the job-ready skills you will graduate with. Please read it fully and keep it for reference throughout your studies.

Over the programme you will build one complete, real-world application — a **Task Management web app** — step by step: a modern React interface, a secure Java Spring Boot REST API and a MySQL database, deployed online and published in your own GitHub portfolio.

Our motto is **Learn · Innovate · Lead**. We teach the way the industry works: hands-on, project-based, with real tools and real code reviews from practising developers.

About SIFT Global DLP

SIFT Global — the **School of Innovation and Future Technologies** — offers industry-aligned technology programmes. Our Distance Learning Programme lets you study from anywhere through live online classes, session recordings and an online learning platform, with the same standards, mentoring and certification as on-campus learning.

Our technology partners

MernCrest Solutions (Pvt) Ltd and **WebFix** are our industry technology partners. They contribute real-project experience to the curriculum and provide the setting for the internship training. Learning resources are delivered through **MernCode Academy** (MernCode.online).

Programme at a glance

Award	Advanced Certificate in Full Stack Development
Awarding body	SIFT Global — School of Innovation and Future Technologies
Mode	Distance Learning Programme (live online + recordings)
Duration	3 months · 24 live classes (8 per month, 2–3 hours each)
Self-study	≈ 4–6 hours per week (labs, assignments, capstone)
Total learning	≈ 150 hours, including the capstone project
Lecturers	K. P. Tashini Kawindi (Frontend) — BSc (Hons) Computing, Coventry University, UK M.B.H.M.U. Pehesara (Backend) — HND in Software Engineering; BSc (Hons) Computing, Coventry University, UK
Course fee	**Rs. 29,000.00** ~~Rs. 35,000.00~~ (special offer)
Capstone	Full Stack Task Management Web Application
Internship	3-month unpaid internship training (eligible graduates)
Language	English (support available in Tamil & Sinhala)

BEFORE YOU ENROL

Is this programme right for you?

School leavers
Students after O/L or A/L who want a practical, job-focused start in software development.

Undergraduates
IT, computing and engineering students who want real project experience to stand out for internships.

Career changers
Working adults moving into tech who need a structured path, mentoring and a portfolio.

What makes this programme different

- **One real project, end-to-end** — every session adds a feature to your capstone.
- **Two specialist lecturers** for Frontend and Backend, in alternating sessions.
- **Industry tools from day one** — Git, GitHub, IntelliJ, Postman, Swagger.
- **Security & deployment included** — JWT login, testing, cloud hosting.
- **Internship pathway** — 3 months of supervised training for eligible graduates.
- **Career-ready portfolio** — live app, GitHub, CV and LinkedIn profile.

Entry requirements

Requirement	Details
Academic	G.C.E. O/L (or equivalent) recommended. No prior programming experience is required, but basic computer skills are essential.
English	Ability to read technical material and follow lectures in English.
Readiness module	All students must complete the self-paced **Module 0 — Readiness** (page 21) before Session 1. It covers HTML, JavaScript and SQL basics and setting up your computer.
Commitment	Attend live sessions (or watch the recording within 48 hours) and spend 4–6 hours per week on labs and assignments.

Equipment you will need

Item	Minimum	Recommended
Computer	Laptop or desktop · Windows 10/11, macOS or Linux	Laptop (useful for the internship)
Processor · RAM	Intel i3 / Ryzen 3 · 8 GB RAM	Intel i5 / Ryzen 5 · 16 GB RAM
Storage	20 GB free space	SSD drive
Internet · Audio	Stable connection for Zoom · microphone or headset	Broadband / fibre · webcam

Phones and tablets are not suitable for coding labs — you need a computer where you can install software. All software used is **free**: JDK 21, Node.js, VS Code, IntelliJ IDEA Community, MySQL, Postman, Git and GitHub.

HOW YOU WILL LEARN

Studying through the Distance Learning Programme

Distance learning at SIFT Global is **live and interactive**: learn with lecturers in real time, then practise with recordings, guided labs and mentor support.

① Live class
2–3 hour Zoom session: concept, live coding demo, guided practical and Q&A.

② Recording
Every session is recorded and uploaded to the learning platform for revision.

③ Lab task
A practical task pushed to your GitHub repository before the next session.

④ Feedback
Lecturer and mentor review of your code, with comments you act on.

A typical week

When	Activity	What you do
Session A	Frontend live class	Learn and build a frontend feature for the capstone.
Between sessions	Lab + self-study (2–3 hrs)	Finish the lab, push to GitHub, watch the recording if needed, post questions in the class group.
Session B	Backend live class	Learn and build the matching backend feature (API, database, security).
Weekend	Practice + mentor support (2–3 hrs)	Integrate frontend and backend, fix bugs, attend optional help-desk / doubt-clearing time.

Exact days and times for your batch are confirmed on enrolment and on the learning platform. **Attendance:** attend at least 80% of live sessions; if you miss one, watch the recording within 48 hours and submit the lab on time — this counts towards attendance.

Your learning tools

Learning platform
MernCode Academy (MernCode.online): recordings, slides, lab sheets, submissions and announcements.

Zoom
Live classes, screen sharing, breakout rooms and the capstone viva. Camera on during practical help.

GitHub
Every lab and the capstone lives in your GitHub repository — your public portfolio from week one.

Class WhatsApp group
Announcements, quick questions and peer help. Be respectful and stay on topic.

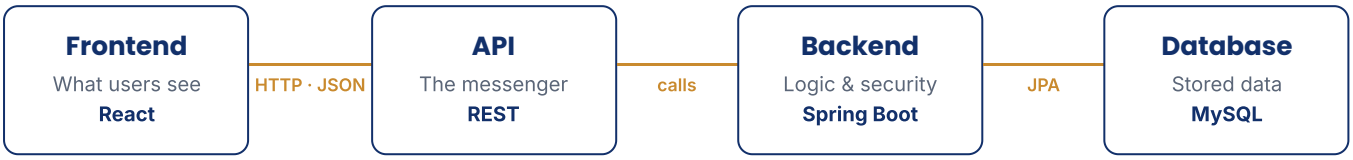
Mentor help-desk
Scheduled online support sessions to unblock errors and review your progress.

AI coding assistants
Taught as a professional skill — use them to learn and debug, never to submit code you cannot explain.

WHAT YOU WILL BUILD WITH

The technology stack

A “stack” is the set of technologies that work together to make one application. You will learn a complete, industry-standard stack — and exactly how each layer connects to the next.



Layer	Technologies	What you use it for
Page structure & design	HTML5 · CSS3 · Flexbox / Grid · Tailwind CSS	Building responsive screens for phone, tablet and laptop
Interactivity	JavaScript ES6+ · fetch / Axios	Clicks, forms, live data and calling the API
Frontend app	React · Vite · React Router	Reusable components, app state, pages and login screens
API	REST · JSON · Swagger / OpenAPI	The contract between frontend and backend, documented
Backend	Java 21 · Spring Boot · Spring Data JPA	Business rules, validation and data access
Security	Spring Security · JWT · BCrypt	Login, protected data and safe password storage
Database	MySQL · SQL	Storing and searching users and tasks
Quality & delivery	JUnit 5 · Mockito · Docker basics · cloud hosting	Testing your code and putting the app online

Tools you will use

Writing code
VS Code · IntelliJ IDEA · Chrome DevTools

Testing & data
Postman · Swagger UI · MySQL Workbench

Teamwork
Git · GitHub · Trello / Jira · AI coding assistants

Why this stack is in demand

React
One of the most widely used frontend libraries; a core skill in most frontend and full stack job adverts.

Java & Spring Boot
The backbone of enterprise, banking, telecom and ERP systems — a stable, long-term career skill.

MySQL
Relational databases and SQL are needed in almost every software role, from developer to QA.

COURSE STRUCTURE

How the programme fits together

Component	Format	What it covers
Module 0 — Readiness	Self-paced, before Session 1	Computer setup, Git & GitHub, HTML / JavaScript / SQL fundamentals.
Frontend Track	11 live sessions	HTML5, CSS3 & responsive design, JavaScript ES6+, React, routing, authentication UI, deployment.
Backend Track	11 live sessions	Java & OOP, Spring Boot, REST APIs, MySQL & JPA, validation, security (JWT), testing, deployment.
Professional Skills	Integrated + self-paced	Agile teamwork, Git workflow, AI-assisted coding, CV, LinkedIn, GitHub portfolio, interview preparation.
Capstone Project	Sessions 23–24 + built across all sessions	Full Stack Task Management Web Application, presented at the Capstone Showcase & Viva.
Internship Training	3 months, after certification	Supervised work on real projects with SIFT Global's technology partners (eligible graduates).

The course runs for 3 months with 8 live classes per month. Sessions 1–22 alternate between tracks — odd-numbered sessions are Frontend, even-numbered sessions are Backend — so each frontend feature is matched by its backend feature the same week. Sessions 23–24 are joint capstone sessions.

Module 0 — Readiness (self-paced, complete before Session 1)

The live course starts at an intermediate level. Module 0 makes sure every student starts with the same foundation. Short recorded lessons and checklists are provided on the learning platform.

0.1 · Set up your developer computer

- Install JDK 21, Node.js LTS, VS Code and IntelliJ IDEA Community
- Install MySQL Server & Workbench, Postman and Git
- Verify installations with a provided checklist

0.2 · Git & GitHub essentials

- Create a professional GitHub account
- `init`, `add`, `commit`, `push`, `pull`, `.gitignore`, README
- Create your course repository

0.3 · Web & JavaScript basics

- How the web works: browser, server, HTTP
- HTML tags, basic CSS, variables, data types, functions, arrays
- Simple DOM events (click handlers)

0.4 · Database & SQL basics

- Tables, rows, columns, primary and foreign keys
- `SELECT`, `INSERT`, `UPDATE`, `DELETE`
- Client → server → database architecture

Readiness check: before Session 1 you will submit a link to your GitHub repository containing a simple styled profile page and a screenshot of your working tool setup. This confirms you are ready to start.

MAIN COURSE · MONTH 1 · SESSIONS 1–8

Foundations: web pages, Java & databases

01 Building Web Pages with HTML

FRONTEND

Semantic layout elements, links and assets, forms, labels, input types and attributes, browser-native validation, accessibility basics (alt text, labels, keyboard use), committing work with Git.

Practical › Build an accessible user registration form and push it to GitHub.

02 Java Basics — Thinking Like a Programmer

BACKEND

Java syntax, primitive types, variables, operators, input/output, if/else and switch, loops, methods and parameters, basic debugging in IntelliJ IDEA.

Practical › Write a console task-tracker that adds and lists tasks.

03 Modern CSS Layouts

FRONTEND

Selectors, cascade and specificity, box model, colours and typography, CSS variables, Flexbox and CSS Grid layouts.

Practical › Style a production-grade product-card grid.

04 Object-Oriented Java

BACKEND

Classes and objects, constructors, encapsulation, inheritance, polymorphism, interfaces, collections (List, Map) and exception handling basics.

Practical › Model a Task class and manage a list of tasks in memory.

05 Responsive Design & Tailwind CSS

FRONTEND

Media queries, mobile-first design, responsive images, an introduction to the Tailwind CSS utility framework and component styling.

Practical › Turn the product-card page into a fully responsive Tailwind layout.

06 Databases, SQL & MySQL

BACKEND

Relational database concepts, tables, primary and foreign keys, normalisation basics, SELECT / INSERT / UPDATE / DELETE, filtering, sorting and JOINs in MySQL Workbench.

Practical › Design the TaskFlow database and run real queries on it.

07 JavaScript Fundamentals

FRONTEND

Variables, data types, functions and arrow functions, arrays and array methods (map, filter, find, reduce), objects and JSON, debugging with Chrome DevTools.

Practical › Build a real-time product filter by price and category.

08 Your First Server with Spring Boot

BACKEND

Spring Boot ecosystem and Spring Initializr, Maven and pom.xml dependencies, project structure, application.properties, core annotations (@SpringBootApplication, @RestController, @GetMapping).

Practical › Scaffold the capstone API project and run a live welcome endpoint.

> **End of Month 1:** you can build and style responsive web pages, write JavaScript and Java programs, design a MySQL database and run your own Spring Boot server.

MAIN COURSE · MONTH 2 · SESSIONS 9–16

Building the app: APIs, JavaScript & React

09 DOM, Events & Modern JavaScript

FRONTEND

DOM selection and manipulation, event listeners, propagation and delegation, ES6+ features (destructuring, spread, modules), localStorage.

Practical › Build an interactive to-do list that saves to the browser.

10 Creating REST APIs

BACKEND

REST principles, resources and URIs, HTTP verbs, @RequestBody, @PathVariable, @RequestParam, status codes (2xx, 4xx, 5xx), JSON serialization, Swagger / OpenAPI documentation.

Practical › Build in-memory task endpoints, test with Postman and document them in Swagger UI.

11 Talking to the Internet — Async JavaScript

FRONTEND

Asynchronous JavaScript, Promises, async/await, try/catch, the fetch() API and Axios, loading and error states.

Practical › Fetch JSON from an external API and render it with loading and error states.

12 Saving Data with Spring Data JPA

BACKEND

Spring Data JPA and Hibernate, datasource configuration, @Entity and @Id, entity relationships (@ManyToOne user → tasks), JpaRepository and derived queries.

Practical › Connect Spring Boot to MySQL; persist and query tasks linked to users.

13 Introduction to React

FRONTEND

Why React, Vite project setup, JSX, component architecture, props, imports/exports, rendering lists, keys and reconciliation, organising a React project.

Practical › Build a reusable task-card and listing component library.

14 Building Every Feature Properly — CRUD

BACKEND

Controller, Service and Repository layers, transactions (@Transactional), pagination, sorting and search with Pageable, clean code conventions.

Practical › Implement full persistent CRUD with pagination and search.

15 Interactive Screens & Forms in React

FRONTEND

useState, controlled inputs, form validation, conditional rendering, immutable updates, lifting state up and reusable custom hooks.

Practical › Build the interactive task list with add / edit modal forms.

16 Checking Data & Handling Errors

BACKEND

Data Transfer Objects, Jakarta Bean Validation (@NotNull, @Size), global exception handling (@ControllerAdvice, @ExceptionHandler), consistent error responses, logging with SLF4J.

Practical › Validate task payloads and return standard error responses.

> **Mid-course checkpoint (after Session 12):** your frontend renders live data from an API, and your backend has tested, documented REST endpoints saving data to MySQL. This checkpoint is assessed (15%).

MAIN COURSE · MONTH 3 · SESSIONS 17–24

Connect, secure, test & launch

17 Connecting Frontend to Backend

FRONTEND

useEffect and side-effects, async data loading, GET / POST / PUT / DELETE from React, loading spinners, error notices and error boundaries, CORS from the browser side.

Practical › Connect the React client to your live Spring Boot API.

18 Login & Security

BACKEND

CORS configuration, authentication vs. authorization, Spring Security filter chain, password hashing with BCrypt, register / login endpoints, issuing and verifying JWT tokens, role-based access.

Practical › Add secure register/login so each user sees only their own tasks.

19 Pages, Navigation & Login Screens

FRONTEND

React Router, shared layouts, nested and protected routes, auth context, storing tokens safely, logout and session expiry.

Practical › Build the login flow and protected dashboard.

20 Testing Your Code

BACKEND

Unit testing with JUnit 5, mocking with Mockito, testing services and controllers, integration testing concepts, test coverage basics.

Practical › Write unit tests for the task service and controller.

21 Publishing Your Frontend Online

FRONTEND

Environment variables (.env), production builds, performance basics, deploying to a static host (e.g. Vercel / Netlify), custom domains overview.

Practical › Deploy the TaskFlow frontend online with a live link.

22 Docker & Cloud Deployment

BACKEND

Spring profiles, managing secrets, JAR packaging, Docker basics (images, containers), deploying the API and MySQL database to a cloud host, end-to-end verification.

Practical › Deploy the TaskFlow API and connect it to the live frontend.

23 Integration & AI-Assisted Development

CAPSTONE

End-to-end integration review, refactoring, AI-assisted development (prompting, reviewing and testing AI-generated code), final bug fixing and README / documentation.

Practical › Finalise TaskFlow, its documentation and demo video.

24 Capstone Showcase & Viva

CAPSTONE

Live demonstration of the deployed application, code walkthrough, viva questions and GitHub portfolio review.

Practical › Present and defend your complete full stack application.

> **Capstone Showcase & Viva (Session 24):** you present your deployed application live, walk through your code and answer questions from the lecturers.

CAPSTONE PROJECT

TaskFlow — your Full Stack Task Management App

From Session 1 to Session 24 you build one application, feature by feature. By the end it is secure, tested, deployed online and ready to show employers.

Frontend — React client

- Responsive task dashboard with status columns
- Login and registration screens with protected routes
- Modal forms to create and edit tasks
- Search, filtering, sorting and pagination
- Loading spinners and friendly error messages

Backend — Spring Boot API

- User and Task domain models with relationships
- Layered architecture with full CRUD endpoints
- Validation, global error handling and logging
- JWT authentication with BCrypt password hashing
- Swagger API documentation and unit tests

Database — MySQL

- Normalised schema with primary and foreign keys
- Indexes for fast searching
- Persistent, per-user task data

Deployment & portfolio

- Frontend and API deployed online with a live link
- Clean GitHub repository with a professional README
- Short demo video for your CV and LinkedIn

Project milestones

Milestone	Frontend	Backend
After Session 8	Responsive pages, JavaScript filter, GitHub repo	Java OOP, database design, Spring Boot server running
After Session 12	Live data loaded from an API with loading & error states	Documented REST endpoints persisting to MySQL
After Session 16	React task list with add / edit forms	Validated CRUD with pagination and error handling
After Session 20	Both tracks integrated: React ↔ API, JWT login working, protected pages, tests written	
Session 24	Deployed, tested full stack application presented at the Capstone Showcase & Viva	

Capstone marking criteria



BEYOND CODE

Professional skills module

Employers hire developers who can work in a team, communicate clearly and keep learning. These skills are woven into the live sessions and supported by short self-paced lessons.

Agile teamwork
Scrum basics, user stories, sprints and stand-ups; managing your capstone on a Trello / Jira board.

Git workflow
Branches, pull requests, code review, merge conflicts and meaningful commit messages.

AI-assisted development
Using AI coding assistants responsibly to learn, debug and review — and checking what they produce.

Technical communication
Writing READMEs and API docs, asking good questions, demoing your work confidently.

Personal branding
A developer CV, an optimised LinkedIn profile and a GitHub portfolio that shows your best work.

Interview readiness
Common technical questions, live coding practice and a mock interview with feedback.

Assessment & certification

Assessment component	Weight
Weekly lab assignments (submitted via GitHub)	**30%**
Mid-course checkpoint — after Session 12	**15%**
Capstone project	**40%**
Capstone Showcase & Viva	**15%**

Grades

Grade	Overall mark
Distinction	75% and above
Merit	65% – 74%
Pass	50% – 64%
Referral	Below 50% — one resubmission opportunity

To earn the Advanced Certificate you must

- ✓ Complete Module 0 — Readiness
- ✓ Attend at least 80% of live sessions (recording + lab counts)
- ✓ Achieve an overall mark of 50% or above
- ✓ Submit a working, deployed capstone project
- ✓ Pass the Capstone Showcase & Viva
- ✓ Meet the academic integrity policy

What you receive

- **Advanced Certificate in Full Stack Development** issued by SIFT Global, with grade
- Verifiable certificate ID
- Eligibility for the 3-month internship training
- On completion of the internship: an internship completion certificate

INDUSTRY EXPERIENCE

3-Month Internship Training

After earning your certificate, eligible graduates continue to **three months of supervised internship training** with SIFT Global's technology partners — working on real software projects alongside professional developers.

The internship turns what you learned in class into real workplace experience: following company coding standards, using team tools, taking tasks from a project board, getting your code reviewed and delivering on deadlines. It is the bridge between "I completed a course" and "I have worked on a real project".

Important — this is an unpaid internship. The internship is a **training placement**. Interns do not receive a salary, stipend or allowance. Its value is the real-world experience, mentoring, internship certificate and professional reference you earn.

HOST PARTNERS

MERNCREST SOLUTIONS

WebFix

Internship roadmap

MONTH **01**

Onboard & Shadow

- Orientation: company tools, code standards, communication and security rules
- Set up the project locally; learn the team's Git branching and pull-request workflow
- Shadow developers; fix small bugs and UI issues; write simple tests
- Daily stand-ups and weekly mentor check-ins

MONTH **02**

Contribute

- Build assigned features: React screens, Spring Boot endpoints, database changes
- Work from user stories on a sprint board; estimate and report progress
- Participate in code reviews — give and receive feedback
- Mid-internship performance review with your supervisor

MONTH **03**

Own & Deliver

- Own a small feature or module end-to-end: design, build, test, document
- Support testing, deployment and release activities
- Prepare handover documentation
- Final internship presentation and evaluation

What you gain

Certificate
Internship completion certificate

Experience
3 months of real project work for your CV

Reference
Supervisor reference for strong performers

Network
Contacts with working developers and teams

INTERNSHIP TRAINING

Eligibility, expectations & evaluation

Who is eligible

- Awarded the Advanced Certificate in Full Stack Development
- Capstone project submitted, deployed and passed
- Good attendance and conduct record during the course
- Successful internship interview / selection discussion
- Signed internship agreement and confidentiality (NDA) form

Placements depend on partner project availability and are allocated on merit. Students are informed of the start date, mode (online / hybrid) and weekly hours in their internship offer letter.

What is expected of interns

- Work the agreed weekly hours and attend stand-ups and meetings on time
- Inform your supervisor in advance of any absence
- Keep company code, data and client information strictly confidential
- Use only company-approved tools and accounts for project work
- Maintain a weekly internship logbook of tasks and learning
- Behave professionally in all communication channels

How interns are evaluated

Technical delivery	30%
Code quality & reviews	20%
Teamwork & communication	20%
Attendance & reliability	15%
Final presentation	15%

Support during the internship

- A named supervisor / mentor from the host partner
- Weekly one-to-one check-ins
- Mid-point and final written feedback
- SIFT Global academic contact for any concerns

> **Please note:** completing the internship does not guarantee employment. Outstanding interns may be considered for future opportunities with partners at their discretion.

Your full journey

1. **Enrol** — Register & receive platform access
2. **Readiness** — Module 0 + tool setup
3. **24 Sessions** — 3 months · Frontend + Backend
4. **Capstone** — Showcase & viva
5. **Certificate** — Advanced Certificate awarded
6. **Internship** — 3 months real projects

CAREERS

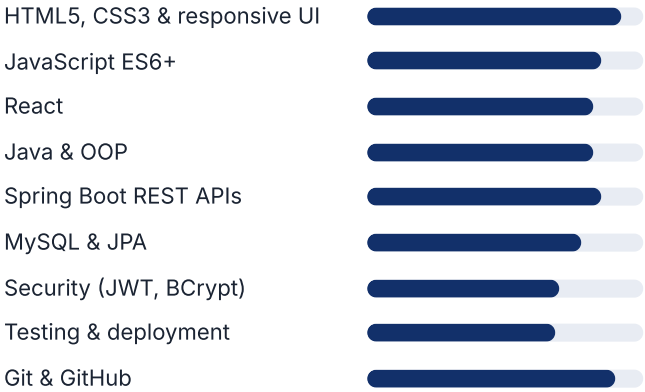
Job-ready skills & career pathways

By the end of the course and internship, you will have the skills employers list in entry-level developer job adverts — and the evidence to prove it.

Roles you can apply for

Role	What you would do	Skills from this programme
Junior Full Stack Developer	Build features across UI, API and database	React, Spring Boot, MySQL, REST, Git
Junior Frontend / React Developer	Build responsive, interactive user interfaces	HTML5, CSS3, JavaScript ES6+, React, Router
Junior Java Backend Developer	Design APIs, business logic and data access	Java, OOP, Spring Boot, JPA, Security, JUnit
Associate Software Engineer	Work in an Agile team on product development	Full stack, Git workflow, Scrum, code reviews
QA / API Test Engineer (trainee)	Test web apps and APIs, report defects	Postman, Swagger, SQL, JUnit, DevTools
Freelance Web Developer	Build web apps for clients locally and online	Full stack build, deployment, client communication

Technical skills you will master



Bars show the relative depth of coverage in the programme.

Workplace skills

Problem solving · Debugging · Agile / Scrum · Teamwork · Code review · Time management · Technical writing · Presenting a demo · Responsible AI use · Self-learning

Your graduate portfolio

- Live, deployed full stack application
- GitHub profile with clean, documented repositories
- API documentation (Swagger) and demo video
- Developer CV and optimised LinkedIn profile
- Advanced Certificate + internship certificate

STUDENT POLICIES

Rules, integrity & frequently asked questions

Academic integrity

All submitted work must be your own. You may discuss ideas and get help, but copying another student's code, submitting code from the internet or AI tools that you cannot explain, or having someone else complete your work is not allowed. Your viva confirms that you understand your project. Breaches can lead to a zero mark or removal from the programme.

Online class conduct

- Join on time with your registered full name
- Keep your microphone muted unless speaking
- Be respectful to lecturers and classmates
- Do not record, share or post class content or links

Submissions & deadlines

Labs are submitted through GitHub before the next session. Late work may be accepted within 48 hours with a mark penalty, unless an extension is approved in advance for a valid reason.

Data & privacy

Your personal details are used only to manage your studies, certification and internship placement.

Frequently asked questions

I have never coded before. Can I join?
Yes. Complete Module 0 — Readiness first; it is designed for beginners.

What if I miss a live session?
Watch the recording within 48 hours and submit the lab. It counts towards attendance.

Can I study while working or at university?
Yes. Sessions are scheduled for working students and every class is recorded.

Is the internship guaranteed?
It is offered to eligible graduates, subject to selection and partner project availability.

Is the internship paid?
No. It is unpaid internship training; the benefit is real experience, mentoring and a certificate.

Can I do the internship online?
Internships are run online or hybrid depending on the project; details are in your offer letter.

Do I need to buy software?
No. All tools used are free. You only need a suitable laptop and internet.

Will I get a job after this?
We prepare you thoroughly — skills, portfolio, CV and interviews — but employment is not guaranteed.

GET READY

Before Session 1 — your checklist

- [ ] I have read this Student Handbook fully
- [ ] I have completed enrolment and received my learning platform login
- [ ] I have joined the class WhatsApp group and checked the timetable
- [ ] My laptop meets the minimum requirements
- [ ] Zoom is installed and my camera and microphone work
- [ ] JDK 21, Node.js, VS Code and IntelliJ IDEA are installed
- [ ] MySQL, Workbench, Postman and Git are installed
- [ ] I have a professional GitHub account and course repository
- [ ] I have completed Module 0 — Readiness lessons
- [ ] I have submitted my readiness check (profile page + setup screenshot)
- [ ] I have planned 4–6 hours per week for self-study
- [ ] I understand the attendance, integrity and internship rules

SIFT GLOBAL
EST. 2026
SCHOOL OF INNOVATION AND FUTURE TECHNOLOGIES
LEARN • INNOVATE • LEAD

We look forward to welcoming you to the next cohort.

LEARN · INNOVATE · LEAD

HOTLINE	WHATSAPP	LEARNING PLATFORM
071 383 8638	**071 383 8637**	**MernCode.online**

TECHNOLOGY PARTNERS

MERNCREST SOLUTIONS

WebFix

Information in this handbook is correct at the time of publication (November 2026). SIFT Global may update course content, schedules and policies to keep the programme aligned with industry needs; students will be informed of any changes.