NEXSKILL

Training • Freelancing • Placement

FORWARD DEPLOYED ENGINEER

MASTER PROGRAM

LEARN. BUILD. DEPLOY.



16 Weeks • 48 Classes • 8 Modules

No-Code • Claude • n8n • Lovable • Python • GitHub • Claude Code & Cursor • LLM APIs • RAG • AI Agents • Docker • Cloud

DURATION

4 MONTHS

Practical • Real-Project Based • Portfolio Focused

3 Classes per Week • Hands-On Labs • Certificate on Completion

INTRODUCTION

ABOUT-THE-PROGRAM

About the Program

The engineer who takes AI from demo to deployed, inside real businesses.

- **What is a Forward Deployed Engineer?:** An FDE works directly with clients to understand their problems, then designs, builds and deploys AI solutions inside their business, from automations and RAG systems to AI agents and full SaaS products.
- **Why This Role?:** Companies everywhere are adopting AI but lack people who can both code and deliver. FDEs combine software engineering, AI tools and client communication, making it one of the fastest-growing roles in tech.
- **What You Will Use:** Claude, n8n, Lovable, Google Stitch & AI Studio, Python, Git & GitHub, Claude Code, Cursor, LangChain, LlamaIndex, LangGraph, AutoGen, Google ADK, MCP, Docker, AWS, Google Cloud and Vercel.
- **Who This Program Is For:** students, developers, freelancers and professionals who want to build real AI systems for clients. It starts with no-code tools, then teaches Python from scratch, so no prior coding is required.

MEET YOUR INSTRUCTORS

INSTRUCTOR · 01

Mian Ahmad Basit

Founder, CEO, CTO & Strategic Advisor



- **Leadership:** Founder, CEO, CTO & Strategic Advisor building AI, Cloud & Blockchain companies and products.
- **Program Architect:** Instructor & Program Architect for Nexskill's advanced AI programs; CTO @ PBI & Anz Tech.

50,000+	13+	5
Talent Developed	Years Executive Leadership	Continents Reached

Connect on LinkedIn ↗ linkedin.com/in/mianahmadbasit

MEET YOUR INSTRUCTORS

INSTRUCTOR · 02

Zeeshan Ali

Technical Project Manager at P2P Clouds

- **Expertise:** Software Engineer & Instructor specializing in AI, Blockchain, Cloud & DevOps.
- **Industry Delivery:** Leads real-world AI & SaaS projects at P2P Clouds for clients across Pakistan & the Middle East.



10,000+ Contributions	**500+** Delivered Projects	**5,000+** Students Trained
6+ Years Experience	**3** Continents Active	

Connect on LinkedIn ↗ linkedin.com/in/zeeshan-ali-dev

CURRICULUM MAP

Program Modules

16 WEEKS · 48 CLASSES · 3 CLASSES PER WEEK

- **Module 1 — Claude & No-Code AI**
  Weeks 1–3 · Classes 1–9
- **Module 2 — Python for AI**
  Weeks 4–6 · Classes 10–18
- **Module 3 — Git, GitHub & AI-Assisted Coding**
  Week 7 · Classes 19–21
- **Module 4 — LLM APIs & Structured Outputs**
  Weeks 8–9 · Classes 22–27
- **Module 5 — RAG Engineering**
  Weeks 10–11 · Classes 28–33
- **Module 6 — Agentic AI & Protocols**
  Weeks 12–13 · Classes 34–39
- **Module 7 — Docker, CI/CD & Cloud Deployment**
  Weeks 14–15 · Classes 40–44
- **Module 8 — Ship, Deliver & Capstone**
  Weeks 15–16 · Classes 45–48

MODULE 1 — CLAUDE & NO-CODE AI · WEEKS 1–3

M1-CLS-1-5

Claude for Productivity

Getting expert results from Claude before writing any code.

- **Class 1 — Meet Claude & Claude.ai:** Anthropic & the Claude model family, choosing models for quality vs speed vs cost, chat, file & data analysis, Projects, memory, web search & deep research.
- **Class 2 — Advanced Prompt Engineering:** clear instructions & context, roles & system prompts, few-shot examples, XML tags, chain-of-thought, prompt chaining, reusable templates.
- **Class 3 — Artifacts, Docs & Design:** interactive artifacts & published apps, Claude Docs & file creation, Claude Design for landing pages, mockups & creatives.
- **Class 4 — Claude in Your Tools:** connectors (Gmail, Drive, Calendar), Claude in Chrome, Excel & PowerPoint, Claude Tag for Slack, permissions & privacy.
- **Class 5 — Agent Skills & Cowork:** SKILL.md structure, custom skills for business processes, Claude Cowork for multi-step knowledge work, repeatable workflows.

MODULE 1 — CLAUDE & NO-CODE AI · WEEKS 1–3

M1-CLS-6-9

No-Code Automation & AI App Builders

Shipping working automations and prototypes for clients in hours, not weeks.

- **Class 6 — n8n Automation:** workflows, triggers & webhooks, connecting APIs, AI nodes with Claude & Gemini, Make & Zapier compared.
- **Class 7 — Lovable:** prompt-to-app development, Supabase for database, auth & storage, adding AI features, custom domains & deployment.
- **Class 8 — Bolt, v0 & Replit:** comparing AI app builders, design-to-code from screenshots & Figma, strengths & limits, avoiding vendor lock-in.
- **Class 9 — Google Stitch & AI Studio:** AI UI design from prompts & sketches, multi-screen flows, Figma & code export, AI Studio playground & app building.

MODULE 2 — PYTHON FOR AI · WEEKS 4–6

M2-CLS-10-14

Python Core Fundamentals

Moving from no-code to code, starting from the very first line.

- **Class 10 — Setup, Syntax & Control Flow:** installing Python & VS Code, variables & data types, operators, if-elif-else, for & while loops, break & continue.
- **Class 11 — Functions & Data Structures:** functions, *args & **kwargs, lambda functions, lists, tuples, sets & dictionaries, comprehensions.
- **Class 12 — Strings, Files & Environments:** string manipulation & regex, file handling, exception handling, modules & packages, venv, poetry & pip.
- **Class 13 — OOP Foundations:** classes & objects, constructors, inheritance (single, multiple, MRO), encapsulation & private attributes.
- **Class 14 — OOP Advanced & Type Hints:** polymorphism & duck typing, abstract base classes, dataclasses & immutability, type hints & static checking.

MODULE 2 — PYTHON FOR AI · WEEKS 4–6

M2-CLS-15-18

Advanced Python & APIs

Writing production-style Python that talks to AI services.

- **Class 15 — Advanced Mechanics:** iterators & the iterable protocol, generators & yield, decorators with state, custom context managers.
- **Class 16 — Async Python:** asyncio event loop, async/await patterns, tasks, gather & semaphores, non-blocking API calls.
- **Class 17 — APIs & Your First LLM Call:** HTTP/REST fundamentals, JSON, environment variables (.env), calling the Claude API from Python, turning an n8n workflow into code.
- **Class 18 — Python Project Structure:** project layout & packaging, logging & configuration, testing with pytest, mini project: a command-line AI assistant.

MODULE 3 — WEEK 7

M3-CLS-19-21

Git, GitHub & AI-Assisted Coding

Version control and the AI coding tools every FDE uses daily.

- **Class 19 — Git & GitHub:** commits, branches & merges, pull requests & code review, resolving conflicts, READMEs & a GitHub portfolio, team collaboration workflows.
- **Class 20 — Claude Code:** setup in terminal, desktop & VS Code, plan mode, exploring codebases, CLAUDE.md, subagents, hooks & slash commands, commits & PRs from Claude Code.
- **Class 21 — Cursor:** AI-first editing, Tab autocomplete, agent mode for multi-file edits, @-mentions & codebase context, Cursor rules & background agents, Cursor vs Claude Code.

MODULE 4 — WEEKS 8–9

M4-CLS-22-27

LLM APIs & Structured Outputs

Building reliable AI features in code with typed, validated responses.

- **Class 22 — Claude API & Platform:** console, API keys & workspaces, Python & TypeScript SDKs, Messages API, streaming, vision & PDF input, token counting.
- **Class 23 — Tool Use & Function Calling:** JSON Schema tool definitions, the tool-use loop, parallel tool calls, server tools (web search, code execution), a tool executor from scratch.
- **Class 24 — Structured Outputs:** Pydantic models as contracts, Instructor with auto-retry, Outlines guided generation, data extraction & classification pipelines.
- **Class 25 — DSPy Programmatic LLMs:** signatures & modules (Predict, ChainOfThought, ReAct), optimizers (BootstrapFewShot, MIPROv2), replacing manual prompt tuning.
- **Class 26 — Gemini API & Multi-Model Strategy:** Gemini SDK, multimodal input, function calling, Claude vs Gemini strengths, routing tasks between models.
- **Class 27 — Cost & Performance:** prompt caching, Batch API for bulk jobs, model choice for cost vs quality, latency optimization, FastAPI backends for LLM features.

MODULE 5 — WEEKS 10–11

M5-CLS-28-33

RAG Engineering

Grounding AI in a client's own documents and data.

- **Class 28 — Embeddings & Vector Search:** dense vs sparse embeddings (bge, e5, BM25, SPLADE), quantized & Matryoshka embeddings, vector databases (Qdrant, Pinecone, PGVector).
- **Class 29 — LangChain:** LCEL pipelines, document loaders & text splitters, retrievers, output parsers, tracing with LangSmith.
- **Class 30 — LlamaIndex:** documents, nodes & chunking strategies, index types, query engines, metadata filtering & hybrid search.
- **Class 31 — Advanced RAG:** hybrid retrieval with reciprocal rank fusion, query rewriting & HyDE, rerankers, contextual retrieval, Self-RAG, Corrective & Agentic RAG.
- **Class 32 — Multimodal & Graph RAG:** document parsers (LlamaParse, Docling), ColPali visual retrieval, Graph RAG with Neo4j, semantic caching with Redis.
- **Class 33 — RAG Evaluation & Security:** RAGAS metrics (faithfulness, context precision & recall), LLM-as-judge, PII masking, prompt injection defense at retrieval.

MODULE 6 — WEEKS 12–13

M6-CLS-34-39

Agentic AI & Protocols

Designing agents that plan, use tools and complete real business tasks.

- **Class 34 — Claude Agent SDK:** the agent loop, workflows vs agents, Agent SDK tools, permissions & sessions, human-in-the-loop approvals, connecting agents to skills.
- **Class 35 — LangGraph Stateful Agents:** StateGraph, nodes, edges & conditional routing, ReAct loops, checkpointers & memory, context management, interrupts, multi-agent supervisors.
- **Class 36 — PydanticAI:** type-safe agents, dependency injection, tool error handling with ModelRetry, structured streaming, Pydantic Evals & Logfire tracing.
- **Class 37 — Microsoft AutoGen:** AgentChat, assistant & user-proxy agents, group chats & agent teams, tool use & code execution, AutoGen Studio, Microsoft Agent Framework.
- **Class 38 — Google Agent Development Kit (ADK):** LLM agents & tools, sequential, parallel & loop workflow agents, multi-agent hierarchies, sessions & state, ADK web UI & evals, deploying to Vertex AI Agent Engine & Cloud Run.
- **Class 39 — MCP & A2A Protocols:** building MCP servers with FastMCP, connecting to Claude, Claude Code & Cursor, MCP security, A2A agent cards & task lifecycle, A2A vs MCP.

MODULE 7 — WEEKS 14–15

M7-CLS-40-44

Docker, CI/CD & Cloud Deployment

Packaging AI apps and running them reliably in the cloud.

- **Class 40 — Docker Fundamentals:** images & containers, writing Dockerfiles, volumes & networking, containerizing a FastAPI + LLM app.
- **Class 41 — Docker Compose & Production Images:** multi-container apps (API, database, vector DB, Redis), multi-stage builds, environment & secrets management, container registries.
- **Class 42 — GitHub Actions CI/CD:** workflows & triggers, automated tests & linting, building & pushing Docker images, Claude Code in GitHub Actions for automated review.
- **Class 43 — AWS Deployment:** IAM, EC2, S3, ECR & ECS Fargate, domains, HTTPS & load balancing, monitoring cloud costs.
- **Class 44 — Google Cloud & Vercel:** deploying containers to Cloud Run, frontends on Vercel, Supabase in production, choosing the right platform for each client.

MODULE 8 — WEEKS 15–16

M8-CLS-45-48

Ship, Deliver & Capstone

Taking AI systems to production and delivering them to real clients.

- **Class 45 — Building & Launching an AI SaaS:** PRD & MVP scope, multi-tenant architecture, payments (Stripe, JazzCash, Easypaisa, KuickPay), launch & collecting feedback.
- **Class 46 — Production Readiness & Client Delivery:** guardrails & prompt injection defense, observability with LangSmith & Langfuse, discovery calls, proposals & pricing, selling on Upwork & Fiverr.
- **Class 47 — Capstone Build:** plan, build & deploy a complete AI solution for a real business, with mentor reviews along the way.
- **Class 48 — Capstone Demo & Certification:** live client-style demo, case study & GitHub portfolio, practical assessment, Forward Deployed Engineer Certificate award.

PROGRAM OUTCOMES

- Automate work and ship prototypes fast with Claude, n8n, Lovable, Bolt, v0, Google Stitch and AI Studio, before writing any code.
- Write production-style Python and use Git, GitHub, Claude Code and Cursor as everyday engineering tools.
- Build reliable AI features with the Claude and Gemini APIs, tool use, structured outputs and DSPy.
- Build production RAG systems with LangChain, LlamaIndex, hybrid search, rerankers and evals.
- Build AI agents with the Claude Agent SDK, LangGraph, PydanticAI, Microsoft AutoGen and Google ADK, connected through MCP and A2A.
- Containerize AI apps with Docker and deploy them to AWS, Google Cloud and Vercel with GitHub Actions CI/CD.
- Deliver secure, production-ready AI solutions to real clients and earn the Forward Deployed Engineer Certificate.

FINAL CAPSTONE

- Work as a forward deployed engineer for a real business: run discovery, then design and build a complete AI solution combining a no-code automation, a RAG knowledge base and an AI agent connected through MCP, containerize it with Docker, deploy it to the cloud with GitHub Actions, and present a live client-style demo to earn the Forward Deployed Engineer Certificate.

ALUMNI SUCCESS STORIES

NEXSKILL-ALUMNI

Success Stories

Real Nexskill students, now building careers in tech.



Muhammad Hashim

Full Stack AI Developer

Age 13 · One of Pakistan's youngest

Full Stack AI · n8n · AI Integration

View LinkedIn Profile



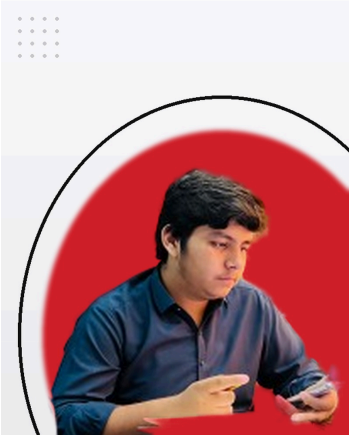
Kashan Iqbal

Senior Software Developer

Flacron Enterprises

Program: Blockchain · Batch B2

View LinkedIn Profile



Faizan Ahmad

Web & AI Developer

P2PClouds

Nexskill Alumni

& Many More...

Nexskill alumni are working as software engineers, AI developers and blockchain developers at companies like Flacron Enterprises and P2PClouds, building their own products, and earning as freelancers on Upwork and Fiverr. The next success story could be yours.

ENROLL NOW Call / WhatsApp 0309-6663330/1 · www.nexskill.com