Akademi

JUNE 2026

GENAI ENGINEERING BOOTCAMP

20-Week Bootcamp Curriculum

LEAD INSTRUCTOR

Gerard Beaubrun

TABLE OF CONTENTS

COURSE OVERVIEW

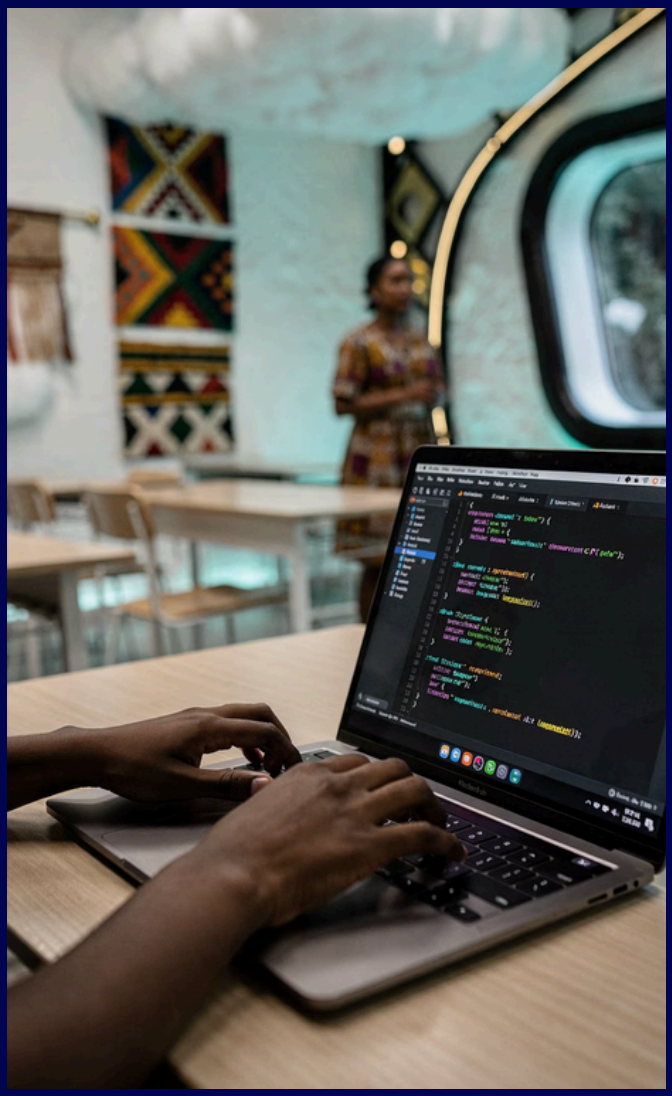
JUNE 2026 - NOVEMBER 2026

AI is only as powerful as the engineer building it—we train you to be that engineer. Akademi's GenAI Engineering Bootcamp equips you with the skills to design, build, and deploy intelligent systems powered by large language models, one of the most in-demand capabilities in today's technology landscape.

The program begins with a strong foundation in Python and core concepts behind large language models, including tokenization and prompt engineering. From there, students dive into advanced topics such as embeddings, Retrieval-Augmented Generation (RAG), vector databases, and agent-based systems. You'll learn how to build production-ready AI applications, integrate tools and APIs, and deploy scalable systems across cloud platforms like Azure and AWS.

Becoming a GenAI engineer requires a blend of software engineering, systems thinking, and practical AI application. This program not only builds those technical skills but also prepares you to navigate a rapidly evolving field—giving you the mindset and tools to continuously adapt and stay ahead in the future of AI.

LEARNING OUTCOMES



Students learn the following skills:

- Build and deploy production-ready applications using large language models (LLMs)
- Design and optimize Retrieval-Augmented Generation (RAG) systems for real-world use cases
- Develop intelligent agents capable of reasoning, tool use, and multi-step workflows
- Work with vector databases, embeddings, and prompt engineering to improve AI performance
- Deploy, monitor, and optimize AI systems in cloud environments with cost, security, and scalability in mind

CURICULUM

The program runs over 20 weeks, combining 80 hours of live instruction with approximately 150 hours of hands-on project work. Students participate in two live sessions per week and complete five portfolio projects designed to build practical, job-ready GenAI engineering skills.

Phase 1	Foundations in Python & GenAI (Weeks 1-4)
Phase 2	Retrieval-Augmented Generation (RAG) Mastery (Weeks 5-10)
Phase 3	Agents & Intelligent Systems (Weeks 11-15)
Phase 4	Cloud Deployment & Optimization (Weeks 16-19)
Phase 5	Capstone Project (Week 20)

PHASE 1: FOUNDATIONS IN PYTHON & GENAI

Week 1: Python Bootcamp + GenAI Overview

Students build a strong programming foundation in Python, covering core concepts such as variables, functions, loops, and data structures. They make their first interaction with a large language model by making a GPT API call, ensuring a technical baseline for the rest of the program.

Week 2: LLM Fundamentals + Tokenization

This week introduces the foundational concepts behind large language models, including how transformers work and how text is processed through tokenization. Students learn how context windows and token usage impact performance and cost by creating a token cost calculator.

Week 3: Prompt Engineering Basics

Students develop critical GenAI skills in prompt engineering, learning zero-shot, few-shot, and chain-of-thought techniques, and designing reusable prompt templates. By the end of the week, they create a library of production-ready prompts for real-world use cases.

Week 4: Embeddings + Vector Similarity

Students learn how text is transformed into numerical embeddings to enable semantic search and retrieval. They explore similarity metrics like cosine similarity and apply these concepts by building a semantic search tool that retrieves the most relevant documents.

PHASE 2: RETRIEVAL-AUGMENTED GENERATION (RAG) MASTERY

Week 5: RAG Architecture Overview

Students are introduced to Retrieval-Augmented Generation (RAG), a key architecture for building practical AI applications. They learn how retrieval, storage, and generation components work together, and why RAG is often preferred over fine-tuning. The week culminates in building a simple RAG pipeline.

Week 6: Document Ingestion + Chunking

This week dives into how documents are processed for use in RAG systems. Students learn various chunking strategies and why they are critical to retrieval quality. They build a document ingestion pipeline capable of handling multiple file types and preparing them for efficient indexing.

Week 7: Vector Databases

Students explore the role of vector databases in storing and retrieving embeddings at scale. They compare different solutions and understand trade-offs related to cost, performance, and scalability. Through hands-on work, they index the same dataset in multiple databases and evaluate performance differences.

Week 8: Retrieval Optimization

This week focuses on improving the quality of retrieval in RAG systems. Students learn evaluation metrics and techniques such as re-ranking to significantly enhance performance. They apply these methods to optimize their systems and achieve measurable improvements over baseline results.

PHASE 2 (CONT): RETRIEVAL-AUGMENTED GENERATION (RAG) MASTERY

Week 9: RAG Evaluation + LangSmith

Students learn how to rigorously evaluate RAG systems using structured metrics and datasets. They build evaluation pipelines to assess retrieval accuracy, answer quality, and relevance. Using tools like LangSmith, they create dashboards to track and improve system performance.

Week 10: Production RAG System (Portfolio Project #1)

In this milestone week, students consolidate their learning by building and deploying a complete RAG system tailored to a real-world domain. They deliver a full-stack solution including an API, frontend interface, and technical documentation, demonstrating production readiness.

Students build a complete, deployed RAG system for their chosen domain.

MEDICAL RECORDS

LEGAL DOCS

CUSTOMER SUPPORT

PHASE 3: AGENTS & INTELLIGENT SYSTEM

Week 11: Agent Fundamentals + ReAct Pattern

Students are introduced to AI agents and how they extend LLM capabilities through reasoning, planning, and tool use. They learn the ReAct framework and build agents that can interact with multiple tools to complete tasks autonomously.

Week 12: Function Calling + Tool Use

This week deepens agent capabilities by teaching students how to define, integrate, and manage tools. They build custom tool interfaces, connect APIs and databases, and ensure robust error handling, resulting in more capable and reliable agents.

Week 13: LangGraph + Stateful Workflows

Students learn to design complex, stateful workflows using LangGraph. They explore how to manage multi-step processes, incorporate conditional logic, and include human-in-the-loop decision points, enabling more advanced and controlled AI systems.

Week 14: Multi-Agent Systems + Orchestration

This week focuses on building systems where multiple agents collaborate to solve complex problems. Students learn orchestration patterns, memory management, and guardrails, and design multi-agent pipelines that mimic real-world team structures.

PHASE 3 (CONT): AGENTS & INTELLIGENT SYSTEM

Week 15: Production Agent System (Portfolio Project #2)
Students develop and deploy a full multi-agent system for a real-world application. This project demonstrates mastery of agent design, orchestration, and deployment, and includes a complete product with documentation and performance analysis.

Build a multi-agent system for a real use case 4 proves mastery of agentic patterns.

CUSTOMER SUPPORT

RESEARCH ASSISTANT

CODE REVIEWER

Required Deliverables:

- Deployed Agent API : Fully functional API exposing all agent capabilities, with secure endpoints, error handling, and usage documentation.
- Frontend Interface : User-friendly interface for interacting with agents, displaying outputs clearly, and supporting real-time interaction.
- Cost Analysis : Detailed breakdown of system infrastructure and operational costs.
- Architecture Writeup : Technical documentation of system design, including agent roles, orchestration, data flow, deployment, and design trade-offs.
- Demo Video : Short video showcasing system functionality, frontend/API interactions, and highlighting performance and real-world application.

PHASE 4: CLOUD DEPLOYMENT & OPTIMIZATION

Week 16: Azure AI Services Deep Dive

Students gain hands-on experience deploying AI systems in the cloud using Azure services. They learn how to manage models, configure search systems, and monitor applications in production environments.

Week 17: AWS Bedrock + Multi-Cloud Awareness

This week expands students' cloud knowledge by introducing AWS services. They deploy systems on AWS and compare it with Azure, developing an understanding of multi-cloud strategies and cost considerations.

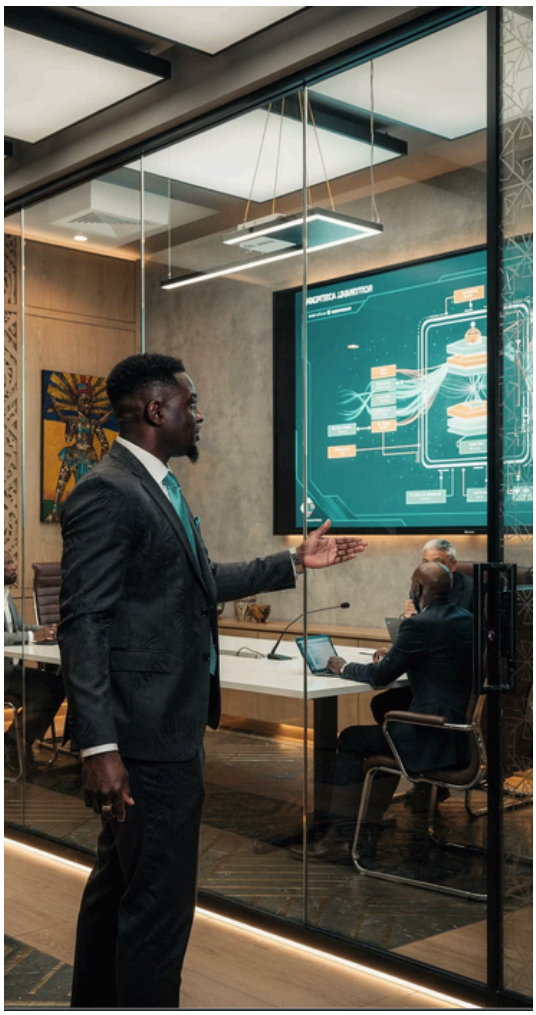
Week 18: Cost Optimization + Production Monitoring

Students learn how to optimize AI systems for cost and performance. They implement techniques such as caching, prompt optimization, and monitoring tools to reduce expenses and improve efficiency in production environments.

Week 19: Security, Governance, Testing

This week focuses on enterprise-grade requirements, including security, compliance, and testing. Students implement safeguards such as PII detection, content filtering, and access controls, while also building automated testing pipelines.

PHASE 5: CAPSTONE PROJECT



Week 20: Capstone Project + Certification

In the final week, students present their capstone projects as if pitching to clients. They demonstrate technical depth, system design, and communication skills through presentations, documentation, and scalable deployment strategies, culminating in program certification.

Present your best system as if pitching to a client.

- 15-Minute Presentation : Architecture, costs, evaluation, security, deployment, roadmap.
- Technical Documentation : Complete system docs + deployment guide .
- Scale Projections : Cost modeling at 1K, 10K, 100K users.
- GitHub Portfolio : 4-5 polished repos ready for internal and external stakeholders.

QUESTIONS? CONTACT US.

At Akademi, we're committed to helping you learn the skills to start or elevate your career. Contact us for any additional information about the GenAI Engineering Bootcamp.

Akademi

www.myakademi.co
info@myakademi.co