genaiexpertise

Remitpro

A Practical Guide in Building Generative AI (GenAI) Applications

Hands-on training to design, develop, deploy, and scale real-world AI applications

www.genaiexpertise.com

Course Overview

This comprehensive, practical guide provides end-to-end training in building Generative AI applications. Over 12 weeks, participants will master Retrieval-Augmented Generation (RAG), FastAPI integration, LangChain and LlamaIndex orchestration, deployment strategies, scaling techniques, security implementation, and monitoring practices. The course combines hands-on labs with real-world application development, culminating in a capstone project that simulates production GenAI deployment.

The program is designed with flexibility in mind, offering online, on-site, or hybrid delivery modes. Each week focuses on building practical competencies through workshop sessions and lab exercises, ensuring participants gain immediately applicable skills. The course structure balances conceptual understanding with hands-on practice, allowing learners to build confidence while developing sophisticated AI applications.

Course Details

- **Duration:** 12 Weeks (3 days/week, 3–4 hours/day)
- **Format:** Hands-on workshop + lab exercises + capstone project
- **Mode:** Online / On-site / Hybrid
- **Delivery:** Interactive sessions with code reviews and Q&A

Target Audience

Data Scientists & ML Engineers

Professionals expanding into GenAI applications

Software Developers

Engineers interested in AI integration

IT Professionals

Technical staff embracing AI transformation

CS/Data Science Students

Advanced learners seeking practical AI skills

Course Objectives

This course delivers comprehensive competencies in Generative AI application development through structured, progressive learning. Each objective builds upon previous knowledge, creating a cohesive learning journey from foundational concepts to advanced deployment strategies. Participants will develop both theoretical understanding and practical expertise, enabling them to tackle real-world AI challenges with confidence.

01

Master GenAI Fundamentals

Understand the core principles of Generative AI and Large Language Models, including their architecture, capabilities, and limitations in real-world applications.

02

Build RAG Applications

Construct and process datasets for Retrieval-Augmented Generation applications, mastering data preparation and optimization techniques.

03

Integrate Vector Databases

Connect vector databases with LLMs for intelligent retrieval, implementing efficient similarity search and document retrieval pipelines.

04

Develop Query Engines

Create sophisticated query engines using LangChain and LlamaIndex, orchestrating complex workflows and prompt chains.

05

Build REST APIs

Develop production-ready REST APIs with FastAPI for AI-driven applications, ensuring robust and scalable service architecture.

06

Deploy with Containers

Containerize AI applications using Docker and Docker Compose, mastering deployment strategies for different environments.

07

Implement Security

Apply security and governance best practices in GenAI applications, including authentication, authorization, and compliance measures.

08

Scale Applications

Design and implement scaling strategies for high concurrency and low latency, optimizing performance under production loads.

09

Monitor Performance

Establish comprehensive monitoring systems to track performance metrics and optimize resource utilization across the application stack.

10

Execute Capstone Project

Demonstrate mastery through an end-to-end GenAI deployment project, integrating all learned skills into a portfolio-ready application.

Learning Outcomes

Upon successful completion of this course, participants will have developed a comprehensive skill set that bridges the gap between AI theory and practical application. These outcomes represent measurable competencies that directly translate to workplace capabilities, enabling graduates to contribute immediately to AI-driven projects and initiatives.

Build RAG Applications

Construct fully functional Retrieval-Augmented Generation applications from scratch, implementing efficient document retrieval and response generation.

Integrate LLM Pipelines

Seamlessly integrate Large Language Models with document retrieval pipelines, creating intelligent systems that leverage both retrieval and generation.

Deploy Scalable Apps

Deploy production-ready AI applications using containerization and orchestration tools, ensuring reliability and scalability.

Ensure Security

Implement comprehensive security measures including role-based access control, authentication, and compliance frameworks.

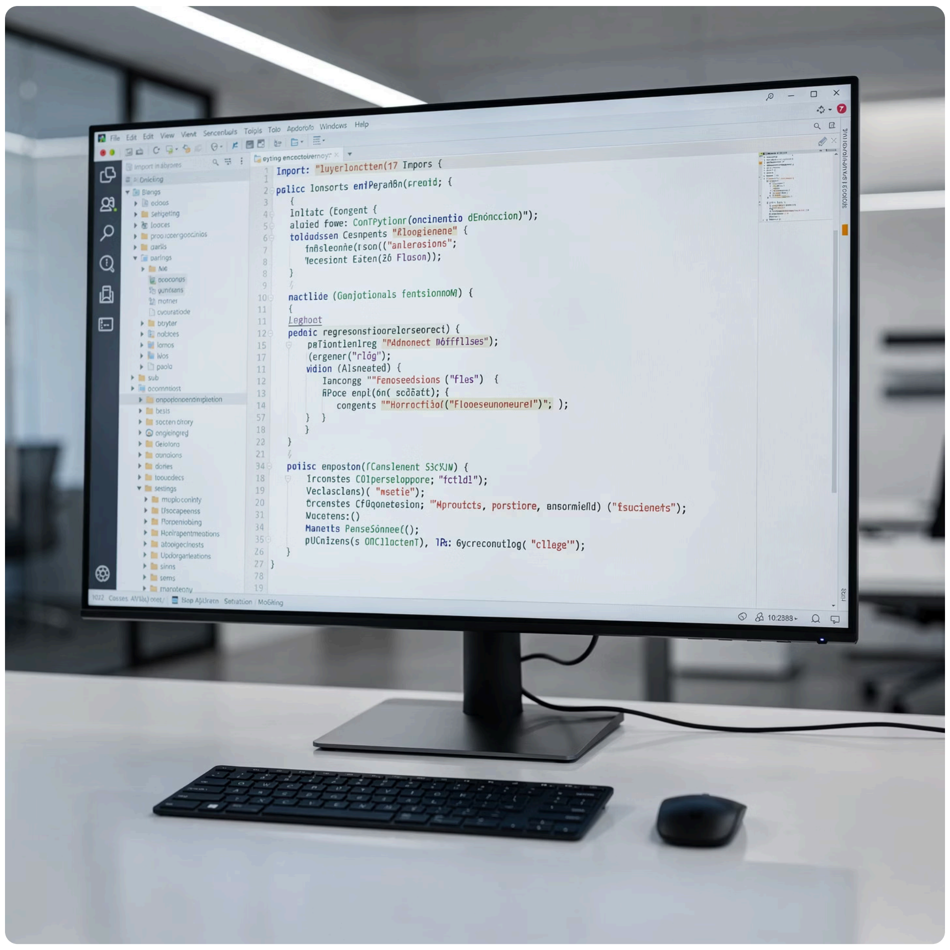
Optimize AI Pipelines

Optimize AI pipelines for cost efficiency, performance, and throughput, balancing quality with resource constraints.

Showcase Expertise

Demonstrate comprehensive skills through a capstone project suitable for professional portfolios and job applications.

Target Competencies



This course develops a carefully curated set of technical competencies that align with industry demands for GenAI professionals. Each competency area has been selected based on real-world application requirements and employer expectations. Participants will build expertise across the full technology stack required for modern AI application development, from foundational programming skills to advanced deployment and monitoring techniques.

The competency framework emphasizes both breadth and depth, ensuring graduates possess well-rounded capabilities. Skills are developed progressively throughout the course, with each module building upon previous competencies to create integrated expertise.

Python & AI Libraries

Advanced Python programming with modern AI libraries including LangChain, LlamaIndex, and transformers ecosystem.

Data Preprocessing

Text and document processing, chunking strategies, and data preparation for embedding generation and retrieval.

Vector Embeddings

Vector embeddings creation, similarity search algorithms, and efficient retrieval system implementation.

API Development

REST API design, microservice architecture patterns, and service-oriented application development.

Containerization

Docker and Docker Compose proficiency with optional Kubernetes orchestration for production deployments.

Security Practices

Application security best practices including authentication, authorization, PII protection, and audit logging.

Observability

Comprehensive logging, metrics collection, monitoring systems, and performance optimization techniques.

Prerequisites

To maximize learning outcomes and ensure all participants can fully engage with course materials, we've established foundational prerequisites. These requirements have been carefully calibrated to be accessible while ensuring participants possess the baseline knowledge necessary for success. The prerequisites are intentionally moderate, allowing professionals from diverse backgrounds to participate while maintaining course rigor and pace.

Python Programming Knowledge

Basic to intermediate understanding of Python programming, including data structures, functions, classes, and common libraries. Participants should be comfortable reading and writing Python code.

API & Web Development Familiarity

While not strictly required, familiarity with APIs, web development concepts, or database systems will help participants grasp concepts more quickly. Understanding of HTTP, REST principles, or SQL is beneficial.

AI/ML Fundamentals Recommended

A general understanding of artificial intelligence and machine learning concepts is recommended but not mandatory. The course provides necessary context, though prior exposure to ML concepts will enhance comprehension.

Note: Don't let prerequisites discourage you! Motivated learners with strong programming fundamentals and dedication to learning can successfully complete this course. Pre-course materials and resources will be provided to help bridge knowledge gaps.

Course Curriculum

The 12-week curriculum follows a carefully structured progression, beginning with foundational concepts and advancing through increasingly sophisticated implementation techniques. Each week focuses on specific competencies while reinforcing previous learning, creating a cohesive educational journey. The curriculum design emphasizes hands-on practice with real-world scenarios, ensuring participants develop practical skills alongside theoretical understanding.

1

Week 1: Introduction to GenAI

Lab 01 – Overview of Large Language Models, GenAI applications landscape, and environment setup with necessary tools and API access.

2

Week 2: Data Ingestion

Lab 02 – Document loading, data cleaning pipelines, text chunking strategies, and storage preparation for embedding generation.

3

Week 3: Vector Storage

Lab 03 – Embedding generation techniques, vector database setup with Weaviate or FAISS, and similarity search implementation.

4

Week 4: RAG Query Engine

Lab 04 – Building retrieval pipelines, implementing basic query engines, and integrating retrieval with generation.

5

Week 5: FastAPI Integration

Lab 05 – REST API development for querying the RAG engine, request handling, response formatting, and error management.

6

Week 6: LangChain + LlamaIndex

Lab 06 – Integrating LangChain for prompt chains, LlamaIndex for query orchestration, and advanced retrieval strategies.

7

Week 7: Multi-Agent Systems

Lab 07 – Parallel query engines, model routing strategies, and orchestrating multiple LLMs for enhanced capabilities.

8

Week 8: Docker Deployment

Lab 08 – Application containerization, Docker Compose for multi-service orchestration, and deployment best practices.

9

Week 9: Monitoring Systems

Lab 09 – Logging implementation, metrics collection, Prometheus and Grafana integration for system observability.

10

Week 10: Security & Governance

Lab 10 – JWT authentication, role-based access control, PII redaction techniques, and comprehensive audit logging.

11

Week 11: Scaling Strategies

Lab 11 – Autoscaling implementation, caching strategies, vector database scaling, and load testing methodologies.

12

Week 12: Capstone Project

Lab 12 – End-to-end GenAI application development integrating all learned components into a production-ready system.

> The capstone project represents the culmination of 12 weeks of learning, where participants integrate RAG, LLM orchestration, API development, containerization, monitoring, and security into a complete, deployable application.

Teaching Methodology

Our teaching methodology emphasizes active learning and practical application over passive content consumption. Every concept is reinforced through hands-on exercises, ensuring participants develop muscle memory alongside theoretical understanding. The approach combines structured guidance with opportunities for creative problem-solving, preparing learners for the ambiguity and challenges of real-world AI development.

Hands-On Labs

Step-by-step guided exercises with comprehensive solutions, allowing participants to build working systems while understanding each component.

Conceptual Lectures

Focused presentations providing theoretical foundations and context before diving into practical implementations.

Project-Based Learning

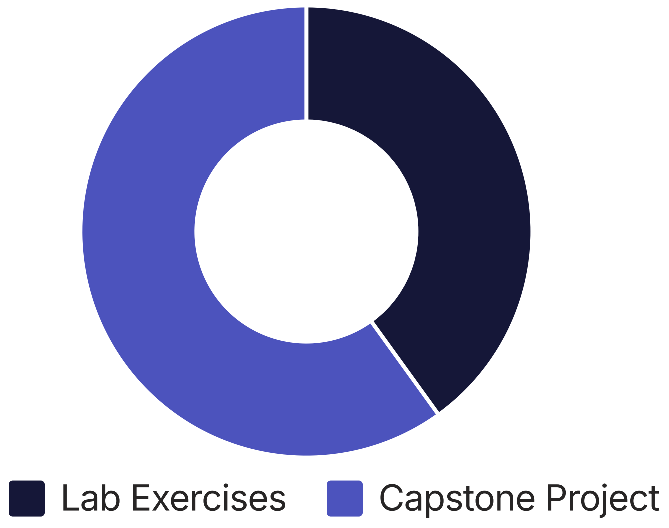
Capstone project integrates all modules, simulating real-world development cycles and deployment challenges.

Interactive Sessions

Regular Q&A discussions, code reviews, and collaborative problem-solving to deepen understanding.

Assessment Structure

Evaluation is continuous and practical, focusing on skill demonstration rather than memorization. Participants receive ongoing feedback throughout the course, allowing for course correction and skill refinement. The assessment approach mirrors real-world job expectations, preparing participants for professional AI development roles.



Tools & Technologies

The course leverages industry-standard tools and technologies that are actively used in production AI systems worldwide. This carefully curated technology stack represents the current state of the art in GenAI application development, balancing cutting-edge capabilities with stability and community support. Participants will gain hands-on experience with each component, understanding not just how to use these tools but when and why to apply them.

Core Programming

Python 3.10+ – Primary development language with modern features and extensive AI library ecosystem.

AI Frameworks

FastAPI, LangChain, LlamaIndex – Web framework, LLM orchestration, and query engine frameworks for building intelligent applications.

Data Storage

Vector Databases – Weaviate or FAISS for embedding storage, with optional PostgreSQL for structured data management.

Large Language Models

OpenAI GPT-4o mini or GPT-3.5 for cloud-based inference, with optional support for local model deployment using open-source alternatives.

Containerization Platform

Docker and Docker Compose for application containerization, environment consistency, and simplified deployment across platforms.

Monitoring & Observability

Prometheus for metrics collection and Grafana for visualization, enabling comprehensive system monitoring and performance optimization.

Security Infrastructure

JWT tokens for authentication, role-based access control systems, and PII filtering mechanisms for data protection.

Delivery & Logistics

We've designed the course delivery model for maximum accessibility and flexibility, accommodating diverse learning preferences and organizational constraints. Whether delivered in-person, online, or through a hybrid approach, the curriculum maintains consistent quality and learning outcomes. The recommended class size balances individual attention with collaborative learning opportunities, creating an optimal educational environment.

Each session is structured to maximize engagement and retention, combining presentation, demonstration, hands-on practice, and discussion. The pacing allows for deep exploration of concepts while maintaining momentum through the curriculum. Regular checkpoints ensure all participants remain on track and can seek clarification as needed.

- **Flexible delivery modes**

  Choose from in-person, fully online, or hybrid format based on your organization's needs and participant locations

- **Optimal session length**

  3-4 hour sessions provide sufficient depth without causing fatigue, with breaks strategically placed throughout

- **Right-sized cohorts**

  15-30 participants allows for personalized attention while fostering peer learning and collaboration

- **Technical requirements**

  Standard development setup with internet connectivity enables full participation from any location

Duration

12 weeks total, with 3-4 hours per session, 3 days per week

Mode

In-person, online, or hybrid delivery options available

Class Size

Recommended 15-30 participants for optimal learning

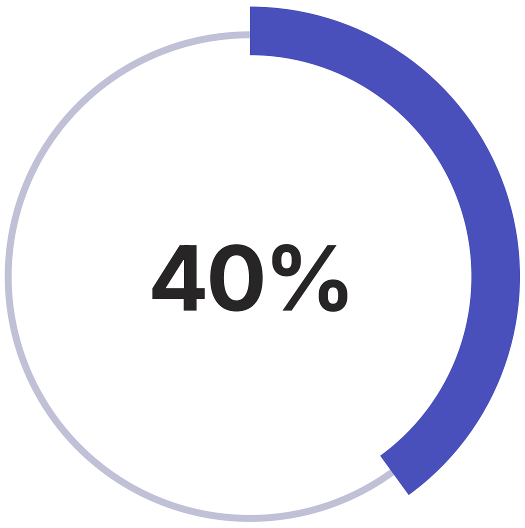
Requirements

Laptop with Python 3.10+, internet connection, OpenAI API key

Assessment & Certification

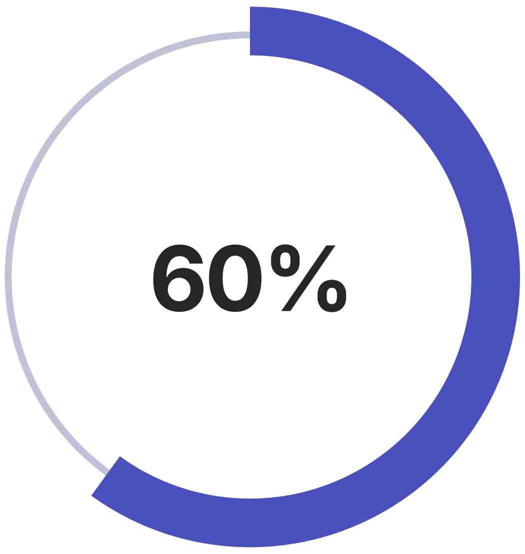
Our assessment framework evaluates both technical competency and practical application ability, ensuring graduates can demonstrate real-world capabilities. The evaluation approach emphasizes skill demonstration through working code and deployable systems rather than theoretical knowledge alone. This practical focus ensures certification represents genuine, market-ready competencies that employers value.

Assessment Components



Lab Exercises

Weekly practical assignments demonstrating progressive skill development



Capstone Project

Comprehensive final project showcasing end-to-end application development

Certification

Upon successful completion of all laboratory exercises and the capstone project, participants receive a **Certificate of Completion** that validates their competencies in building production-ready Generative AI applications.

The certificate serves as tangible proof of skills for career advancement, portfolio building, or academic credit where applicable. It represents mastery of the complete GenAI application development lifecycle.

01

Complete Weekly Labs

Submit working solutions for all 12 laboratory exercises, demonstrating progressive mastery of course concepts.

02

Build Capstone Project

Develop a comprehensive GenAI application integrating all learned components into a production-ready system.

03

Present & Demonstrate

Showcase your capstone project through a technical demonstration, explaining architecture and implementation decisions.

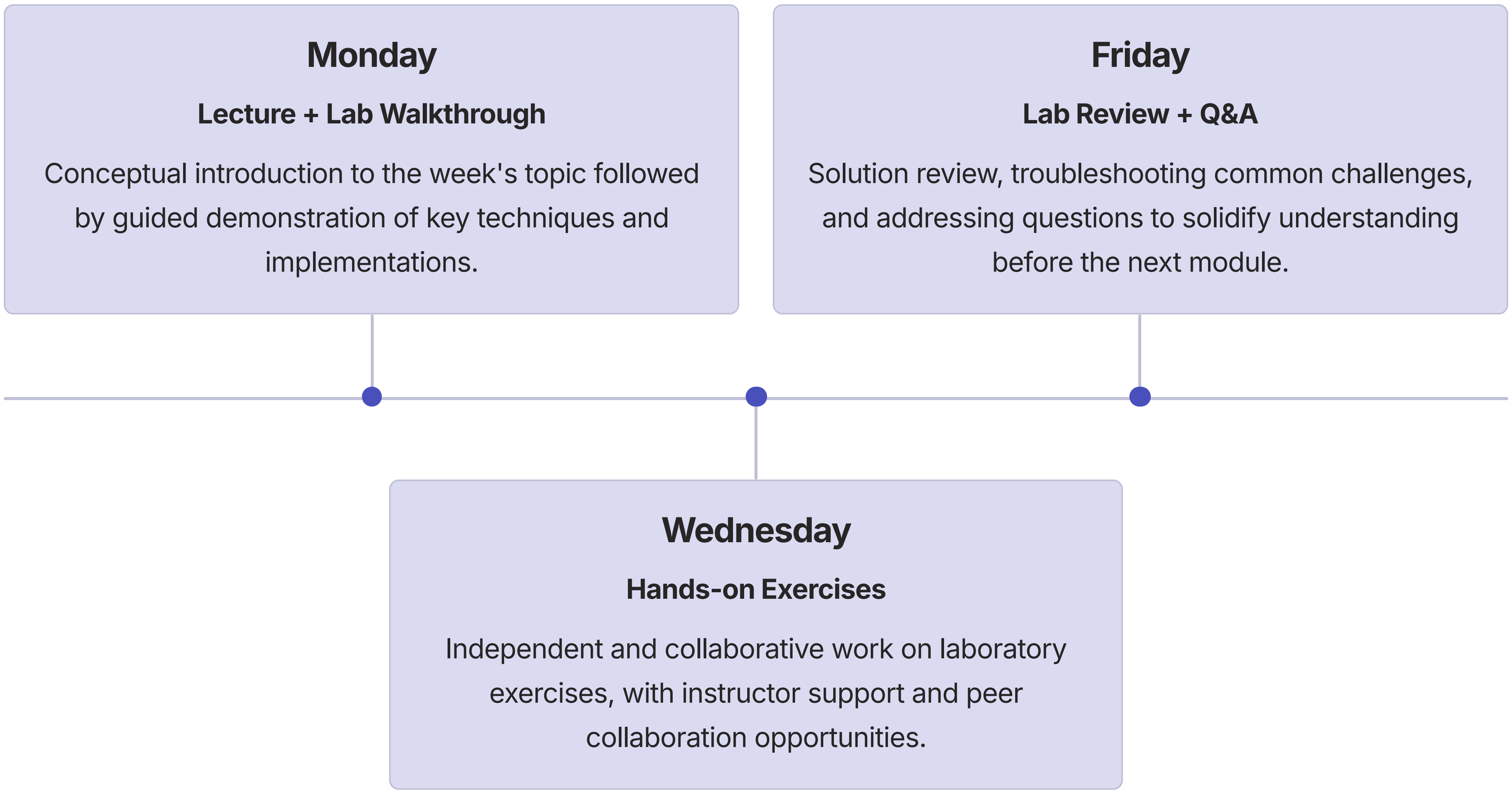
04

Receive Certification

Earn your Certificate of Completion, validating your expertise in GenAI application development.

Proposed Schedule Example

The weekly schedule structure creates a rhythm that optimizes learning retention and skill development. Each week follows a consistent pattern, allowing participants to develop routines while progressing through increasingly sophisticated content. The three-session-per-week cadence provides sufficient practice time between sessions while maintaining momentum and engagement throughout the 12-week program.



Monday

Lecture + Lab Walkthrough

Conceptual introduction to the week's topic followed by guided demonstration of key techniques and implementations.

Friday

Lab Review + Q&A

Solution review, troubleshooting common challenges, and addressing questions to solidify understanding before the next module.

Wednesday

Hands-on Exercises

Independent and collaborative work on laboratory exercises, with instructor support and peer collaboration opportunities.

Weekend Project Hackathon: During weeks 11-12, optional weekend sessions provide extended time for capstone project development, offering opportunities for intensive work, collaboration, and instructor guidance on complex implementation challenges.

Session Structure

Each 3-4 hour session is carefully structured to maintain engagement and maximize learning effectiveness. Sessions include presentation of concepts, live coding demonstrations, hands-on practice time, and collaborative problem-solving. Regular breaks prevent fatigue while allowing for informal discussions and networking.

Between Sessions

Participants are encouraged to experiment with concepts beyond required exercises, explore additional resources, and begin thinking about capstone project ideas. Office hours and discussion forums provide ongoing support between scheduled sessions.

Expected Outcomes for Organizations & Schools

Organizations and educational institutions investing in this training program can expect significant returns through enhanced team capabilities and competitive advantages in AI adoption. The program addresses the critical skills gap in GenAI application development while providing immediate practical value through portfolio-ready projects. Graduates emerge prepared to lead AI initiatives, reducing the need for external consultants and accelerating time-to-market for AI-driven products and services.

Hands-On GenAI Expertise

Staff and students gain practical, immediately applicable skills in building, deploying, and scaling Generative AI applications using industry-standard tools and frameworks.

Deployment Capabilities

Teams acquire the ability to deploy secure, scalable AI applications in production environments, complete with monitoring, security, and governance best practices.

Reduced Learning Curve

Structured curriculum dramatically shortens the path from AI concept to production deployment, eliminating common pitfalls and accelerating productization timelines.

AI-Ready Workforce

Develop a workforce prepared to tackle AI-driven projects with confidence, capable of architecting solutions and making informed technology decisions.

Portfolio-Ready Projects

Participants complete capstone projects suitable for professional portfolios, providing tangible demonstrations of capability for career advancement or recruitment.

Competitive Edge

Organizations gain strategic advantages through internal AI expertise, faster innovation cycles, and reduced dependence on external vendors and consultants.

> This program transforms theoretical AI knowledge into practical capability, enabling organizations to move from AI experimentation to production deployment with confidence and speed.

Proposed Budget & Pricing

Investment in this training program represents a strategic decision to build internal AI capabilities rather than relying on external resources. The pricing structure is designed to be transparent and customizable based on organizational needs, delivery format, and scale. All course materials, including comprehensive lab manuals, code repositories, datasets, and reference documentation, are included in the base price, ensuring participants have everything needed for success.

Additional costs may apply for cloud computing resources if utilizing GPU-accelerated infrastructure for model training or hosting, though many labs can be completed using free-tier cloud services or local development environments. Organizations can choose to provide API access centrally or have participants use individual accounts based on their preferences and policies.

Instructor Fee

Customizable per hour or per session rate based on delivery format, class size, and instructor expertise level. Contact for detailed pricing.

Course Materials

Comprehensive lab manuals, complete code repositories, curated datasets, and reference guides are included with enrollment at no additional charge.

Platform Access

Virtual classroom platform (Zoom or equivalent) for online delivery, with classroom rental costs for in-person sessions determined by venue selection.

Optional: Cloud Resources

Cloud GPU compute costs or LLM API access fees vary based on usage patterns. Free-tier options available for most lab exercises to minimize costs.

Flexible Pricing: All pricing components are customizable to meet organizational budgets and requirements. Volume discounts available for multiple cohorts. Contact us to discuss a pricing structure that works for your specific needs and constraints.

References & Resources

This course builds upon the documentation, best practices, and community knowledge from leading organizations and open-source projects in the GenAI ecosystem. Participants will develop familiarity with these essential resources throughout the program, learning how to leverage official documentation, community forums, and example repositories for continued learning beyond the course.

OpenAI Platform

Official API documentation, usage guides, and best practices for GPT models and embeddings.

LangChain

Comprehensive framework documentation for building LLM-powered applications and chains.

langchain.com

LlamaIndex

Query engine and data framework documentation for connecting LLMs with external data sources.

gpt-index.readthedocs.io

Weaviate

Vector database documentation covering schema design, indexing strategies, and similarity search implementation.

weaviate.io

FastAPI

Modern Python web framework documentation with focus on async APIs and automatic OpenAPI generation.

fastapi.tiangolo.com

Additional resources including research papers, tutorials, blog posts, and community examples will be provided throughout the course to support specific topics and advanced exploration. Participants will also gain access to a curated list of GitHub repositories demonstrating production GenAI implementations and architectural patterns.