FullStack Generative AI Developer

12-Week Course Syllabus

Offered by **Vipracube AI Technologies**

Python | FastAPI & SQL | ML & Deep Learning | LangChain & LangGraph

Duration	12 weeks (4 + 2 + 6)
Prerequisites	Basic computer literacy and logical thinking. No prior programming experience needed.
Outcome	Build, evaluate, and deploy full-stack AI applications: backend APIs, ML models, RAG systems, and multi-agent workflows.

Course Roadmap

Module	Topics	Weeks
1. Python & Backend Development	Python basics, OOP, FastAPI, SQL	1 - 4
2. Machine Learning & Deep Learning	ML fundamentals, neural networks, model serving	5 - 6
3. Generative AI	LLMs, LangChain, RAG, Agents, LangGraph, deployment, capstone	7 - 12

Module 1: Python Foundations & Backend Development

Weeks 1 - 4

Week 1: Python Basics

- Setting up the environment: Python, VS Code, virtual environments, pip, Git/GitHub basics
- Variables, data types, operators, input/output
- Control flow: conditionals and loops
- Data structures: lists, tuples, sets, dictionaries
- Functions: arguments, return values, lambda, scope

Lab / Project: Console-based utilities (calculator, to-do manager)

Week 2: Intermediate Python & OOP

- File handling, exception handling, working with JSON/CSV
- Modules, packages, and the standard library
- List/dict comprehensions, generators, decorators
- OOP: classes, objects, inheritance, polymorphism, encapsulation, dunder methods
- Type hints and dataclasses

Lab / Project: Object-oriented Library or Inventory Management system

Week 3: APIs with FastAPI

- How the web works: HTTP, REST, JSON, status codes
- FastAPI setup, routing, path and query parameters
- Request/response models with Pydantic, validation
- Dependency injection, middleware, error handling
- Async/await basics, background tasks
- Testing APIs with Swagger UI, Postman, and pytest

Lab / Project: Build a CRUD REST API (in-memory)

Week 4: SQL & Database Integration

- Relational database concepts, PostgreSQL/SQLite
- SQL: SELECT, WHERE, JOINs, GROUP BY, subqueries, indexes
- Database design, normalization, primary/foreign keys
- Connecting FastAPI to a database with SQLAlchemy (ORM) and Alembic migrations
- Authentication basics: password hashing, JWT

Lab / Project: Mini Project 1 - Backend for a Task/Blog/Expense app with auth, FastAPI, and SQL

Module 2: Machine Learning & Deep Learning

Weeks 5 - 6

Week 5: Introduction to Machine Learning

- ML overview: supervised, unsupervised, and reinforcement learning
- NumPy, Pandas, Matplotlib for data handling and visualization
- Data preprocessing: missing values, encoding, scaling, train/test split
- Regression: linear and logistic
- Classification and clustering: decision trees, random forest, KNN, K-Means
- Model evaluation: accuracy, precision, recall, F1, confusion matrix, cross-validation, overfitting

Lab / Project: House price prediction and customer churn classification using scikit-learn

Week 6: Introduction to Deep Learning

- Neural network fundamentals: neurons, activation functions, loss functions, backpropagation, gradient descent
- Building networks with PyTorch (or TensorFlow/Keras)
- CNNs (image classification) and RNNs/LSTMs (sequence data): conceptual overview
- Introduction to embeddings and the Transformer architecture (bridge to GenAI)
- Saving models and serving predictions through a FastAPI endpoint

Lab / Project: Mini Project 2 - ML/DL model (e.g., sentiment or image classifier) exposed as an API

Module 3: Generative AI with LangChain & LangGraph

Weeks 7 - 12

Week 7: Generative AI & LLM Fundamentals

- How LLMs work: tokens, context windows, temperature, sampling
- Overview of model providers (OpenAI, Anthropic, Google, open-source via Hugging Face/Ollama)
- Working with LLM APIs directly
- Prompt engineering: zero-shot, few-shot, chain-of-thought, system prompts
- Structured outputs (JSON mode, Pydantic schemas)
- API key management, cost, and rate limits

Lab / Project: Prompt-driven text summarizer/classifier served through FastAPI

Week 8: LangChain Core

- LangChain architecture and components
- Chat models, prompt templates, and output parsers
- LCEL (LangChain Expression Language): chains, runnables, parallelism, streaming
- Conversation memory and message history
- Document loaders (PDF, web, CSV) and text splitters

Lab / Project: Multi-step chains, including a chatbot with memory

Week 9: Retrieval-Augmented Generation (RAG)

- Why RAG: hallucinations, knowledge cutoffs, private data
- Embeddings and semantic search
- Vector databases: FAISS, Chroma, Pinecone/Qdrant
- Chunking strategies and retrievers
- Building an end-to-end RAG pipeline

Lab / Project: "Chat with your PDFs/documents" application

Week 10: Advanced RAG, Tools & Agents

- Advanced RAG: hybrid search, re-ranking, query rewriting, multi-query retrieval, metadata filtering
- Tool/function calling and custom tools
- Agent concepts: ReAct pattern, reasoning and acting loops
- Integrating external APIs, web search, databases, and code execution as tools

Lab / Project: A tool-using agent (e.g., research assistant or SQL-querying agent)

Week 11: LangGraph & Agentic Workflows

- Why LangGraph: limitations of linear chains, graph-based orchestration
- State, nodes, edges, and conditional routing
- Loops, cycles, and self-correction (reflection patterns)
- Persistence and checkpointing, short- and long-term memory
- Human-in-the-loop: interrupts, approvals, resuming workflows
- Multi-agent patterns: supervisor, hierarchical, and collaborative agents

Lab / Project: Multi-agent workflow (e.g., research, write, and review pipeline)

Week 12: Evaluation, Deployment & Capstone

- Observability and tracing with LangSmith
- Evaluating LLM apps: RAG metrics, test datasets, guardrails, prompt-injection basics
- Deploying: Docker, environment configuration, cloud deployment (Render/AWS/Azure)
- Simple frontend integration (Streamlit or basic React/HTML) to complete the "full stack"
- Capstone demos, presentations, and feedback

Lab / Project: Capstone Project - Build, deploy, and present a full-stack AI application

Capstone ideas: AI customer support agent with RAG; resume/job-matching assistant; document Q&A platform with authentication; multi-agent research assistant; SQL-powered business analytics chatbot.

Assessment Structure (Suggested)

Component	Weightage
Weekly assignments / labs	30%
Mini Projects 1 & 2	20%
Mid-course quiz (end of Week 6)	10%
Capstone project	40%

Tools & Tech Stack

Python, FastAPI, PostgreSQL/SQLite, SQLAlchemy, scikit-learn, PyTorch, LangChain, LangGraph, LangSmith, ChromaDB/FAISS, Docker, Git/GitHub, Streamlit.

Enroll / Contact

Vipracube AI Technologies

For more details, please contact: **9960923965**