ComponentSoft
open cloud trainings and consultation

AI-452: Agentic AI Application Development with LLMs (short version)

Course Length: 16 training hours

Course Description:

The fast pace of development in LLMs and related technologies made it possible to use them even in enterprise grade applications. There are already a few areas where a new generation of LLM-based applications totally redefined applications' capabilities and users' expectations while AI technologies are going to radically change all kinds of other software areas as well.

That's why software developers as well as other IT professionals and technical managers need to understand these technologies, especially agentic AI, and need to have practical skills to use them in their daily work.

Training objectives: At the end of the training participants will:

- Write simple programs using both open- and closed-source LLMs, either through their own APIs or with popular frameworks like LangChain.
- Understand the main ideas behind prompt engineering, including practical tips and best practices for working effectively with modern LLMs in chatbots and agentic applications.
- Grasp the fundamental ideas behind Retrieval-Augmented Generation (RAG) systems and apply both its basic and more advanced versions in LLM-based agents.
- Understand the motivations for and the two main types of LLM-based agentic systems (workflows and autonomous agents) as well as the key components and the way of working of autonomous agents.
- Recognize the importance of tracing LLM-based applications throughout their lifecycle and get hands-on experience with tools like LangSmith for tracing.

Main topics:

- Using closed- and open-source LLMs via APIs and LangChain
- Prompt engineering in chatbots and agentic applications
- Retrieval Augmented Generation (RAG)
- LLM-based Agentic Systems
- Workflows, Multi-agent Systems and Agentic Frameworks (optional)
- Tracing and Evaluating LLM-based apps

Structure: 50% theory, 50% hands on lab exercises

Target audience: Software developers, testers, devops as well as other IT professionals and technical managers with technical backgrounds who want to gain practical skills in creating agentic AI applications with LLM APIs and open-source frameworks in Python or other programming languages.

Prerequisites: Basic understanding of AI concepts, basic Python programming skills, user experience with ChatGPT or similar chatbots.

This training is part of the AI portfolio of Component Soft which explores essential AI topics, such as:

- AI-101: Intro to GenAI with Large Language Model (LLMs) and LLM-based apps.
- AI-242: Using Github Copilot and spec-kit for agentic coding and spec-driven

development (SDD)
- AI-262: Using Amazon Kiro for agentic coding and spec-driven development (SDD)
- AI-453: Agentic AI Application Development with LLMs

Detailed Course Outline

Module 1. Using closed- and open-source LLMs via APIs and Langchain

- Using LLMs through APIs
- Typical LLM parameters
- Using LLMs via Langchain
- Creating simple chatbot agents with Langchain
- Labs: Using popular closed- and open-source LLMs via their own APIs and via Langchain, Creating simple chatbot agents with Langchain

Module 2. Prompt engineering for chatbots and agents

- The 4 golden rules of prompt engineering
- 10 Prompting rules of thumb
  - Be concise and give clear instructions
  - Be specific and include relevant details
  - Add positive and negative prompts
  - Define roles for the LLM
  - Define roles for the LLM's audience
  - Provide examples for the solution or response style
  - (one-shot or few-shot prompting)
  - Add relevant context
  - Divide difficult tasks into subtasks (Prompt Chaining)
  - "Let's think step by step" (Chain of Thought)
  - Let LLM ask questions
- **Lab:** Prompt engineering tasks

Module 3. Retrieval Augmented Generation (RAG)

- What is Retrieval Augmented Generation (RAG)
- How do RAG systems basically work?
- Implementation details
- **Lab 1:** Creating simple agentic RAG systems

Module 4. Basics of Agentic Systems

- Motivations for LLM-based Agentic Systems
- Main Features of and Difference between LLM Workflows and Agents
- Main Building Blocks: Functions, Tools, Agents
- The ReAct autonomous agent execution logic
- Implementing Functions, Tools and the ReAct agent execution logic with LangChain
- **Lab:** Creating and using simple LangChain autonomous agents

Module 5. Workflows, Deep Agents, Multi-agent systems and Agentic Frameworks

- Problems with the ReAct model
- First solution: workflows
- Second solution: multi-agent systems
- Third solution: deep agents

- Most popular agentic frameworks

Module 6. Observing (tracing) LLM-based agents

- Why do we need it during development?
- Debugging LangChain-based programs without any monitoring software
- Debugging and evaluation tools for LLM-based apps
- Introducing and Initializing LangSmith
- LangSmith tracing primitives
- Tracing: using LangSmith without and with LangChain
- **Lab:** LangSmith Tracing