Lab 1 – Environment and Solution Setup for Agentic Underwriting AI

This lab prepares learners with a complete development and execution environment for the agentic underwriting AI solution. Participants will set up local development tools, clone the solution repository, configure cloud resources, and validate access to Microsoft Fabric, Azure AI services, and the underwriting solution codebase.

Lab Goals

- Understand the overall solution structure and lab progression
- Set up local development prerequisites for backend and frontend components
- Clone the agentic underwriting solution repository using GitHub Desktop
- Explore the repository structure, including frontend, backend, agent, and infrastructure components
- Validate access to Microsoft Fabric, Azure AI Search, and Azure OpenAI
- Verify connectivity to cloud resources using managed identity or developer credentials

Hands-On Activities

- Install required local tools (Python, Node.js, VS Code, GitHub Desktop)
- Clone the agentic underwriting solution repository
- Review solution folder structure and key configuration files
- Authenticate to Azure using developer credentials
- Run the frontend service locally and validate startup logs
- Execute basic test requests to confirm service connectivity

Dependencies and Prerequisites

- Azure subscription with Contributor access
- Microsoft Fabric workspace (Contributor access)
- Azure OpenAI resource access
- Azure AI Search service
- GitHub account and GitHub Desktop installed
- Local development machine with Python 3.11+, Node.js, and VS Code

Outputs of This Lab

- Local development environment configured and validated
- Agentic underwriting solution repository cloned and understood
- Verified access to required Azure and Fabric services
- Frontend service running locally and ready for further labs

Hands-On Activities: Step by step instructions

1. Clone the agentic underwriting solution repository
    a. Repo link: https://github.com/amulchapla/agentic-underwriting
2. Review solution folder structure and key configuration files
    a. This solution has two components:
        i. Frontend (web UI): Code for frontend component is in agentic-underwriting-ui folder. Frontend is based on Node.js/Next.js stack.
        ii. Backend (API/agents): Code for backend component is in agentic-underwriting-backend. Backend is a Python application
3. Install required local tools (Python, Node.js, VS Code, GitHub Desktop)
    a. For frontend UI: DownloadNode.js/Next.js (or similar)
        i. Download Node.js LTS from https://nodejs.org/
        ii. Run the installer with default settings.
        iii. Change directory to agentic-underwriting-ui
        iv. Install required packages by running the command “npm install -all”
    b. For backend:
        i. Install Python. Download Python from https://www.python.org/downloads/
        ii. Change directory to agentic-underwriting-backend
        iii. Install required dependencies by running the command “pip install -r requirements.txt”
    c. Download and install VS code - Download from https://code.visualstudio.com/
    d. Download and install GitHub desktop. Or you can also download the repo code using a zip file.
4. Run the frontend service locally and validate startup logs
    a. Change directory to agentic-underwriting-ui
    b. Run command “npm run dev”. Wait for the frontend app to start up. Note the local URL for the web UI component.
    c. Go to http://localhost:3000/. You should see the UI component unless there are startup errors.
    d. Use GitHub copilot chat to debug and fix any frontend UI startup errors.
5. Verify contributor access to Microsoft Fabric - Create the Microsoft Fabric Workspace
    a. Sign in to https://app.fabric.microsoft.com using your organizational credentials.
    b. In the left-side navigation panel, select Workspaces → New workspace.
    c. Enter:
        - Name: uw-agentic-ai
        - Description: Workspace for Underwriting Agentic AI application
    d. Click Save.
    e. Verify the workspace appears in your list.
    f. Checkpoint: You should now have a workspace titled uw-agentic-ai.
6. Verify contributor access to Microsoft Azure
    a. Login to portal.azure.com using your organizational credentials.
    b. Create a new resource group name “uw-agenticai-rg”
    c. Create a new Azure AI Search service.
    d. Create a new Microsoft Foundry resource