AI for Engineers Fellowship Setup Guide for Mac OS Computer

Before You Start

What You Need:

- A Mac computer running macOS Ventura (13.0) or later.
- An internet connection.
- A Google account (for Google Colab, Step 8).
- Your GitHub account username and password (you'll need the Microsoft Authenticator app on your phone for 2FA).

Time Needed: About 2–3 hours, depending on your internet speed.

Important Note: If anything doesn't work, contact the fellowship's Tech Support (Shivaraj, email: hi@outskill.com, or raise a ticket on outskill.com).

Step 1: Install Python

1. Open your web browser (e.g., Safari, Google Chrome) and go to python.org/downloads.
2. Look for Python 3.13.4 (or the latest 3.13 version). Click "Download Python 3.13.4" for macOS (choose the "macOS 64-bit universal2 installer").
3. Run the downloaded file (e.g., python-3.13.4-macosx11.pkg in your Downloads folder).
4. In the installer window:
    - Follow the prompts to install (it will install to /Library/Frameworks/Python.framework/Versions/3.13).
    - Ensure "Add Python 3.13 to PATH" is selected (this adds Python to your system's PATH via /usr/local/bin).
5. Wait for the installation to finish, then click "Close."
6. Open Terminal (search for "Terminal" in Spotlight or find it in Applications > Utilities):
    - Type python3 --version and press Enter. You should see Python 3.13.4.
    - Type pip3 --version and press Enter. You should see something like pip 25.1.1.

Step 2: Install Visual Studio Code (VS Code)

1. Go to code.visualstudio.com/download in your browser.
2. Click “Download for Mac” (e.g., VSCode-darwin-universal.zip).
3. Open the downloaded file, drag the Visual Studio Code app to your Applications folder.
4. Launch VS Code from Applications or Spotlight.
5. In VS Code, install these extensions (they help you code in Python):
    - Click the Extensions icon on the left sidebar (or press Cmd+Shift+X).
    - Search for and install:
        - “Python” (by Microsoft).
        - “Python Debugger” (by Microsoft, might be included with Python extension).
        - “Jupyter” (by Microsoft).
    - You’ll see a notification when each extension is installed.

Step 3: Set Up Your Project Folder and Virtual Environment

1. Create a project folder:
    - Open Finder and go to your home directory (/Users/YourUsername).
    - Create a new folder called ai (right-click > New Folder > name it ai).
    - Inside ai, create another folder called ai_fellowship (so the path is /Users/YourUsername/ai/ai_fellowship).
2. Open Terminal:
    - Navigate to your project folder by typing:

cd ~/ai/ai_fellowship

Press Enter.

3. Create a virtual environment (this keeps your project separate from other Python projects):
    - In Terminal, type:

python3 -m venv venv

Press Enter. This creates a folder called venv in /Users/YourUsername/ai/ai_fellowship.

4. Activate the virtual environment:
    - In Terminal, type:

source venv/bin/activate

Press Enter. Your prompt should change to (venv).

5. Install the required Python packages:
    - In Terminal (with (venv) active), type:

pip install streamlit gradio

Press Enter. This installs Streamlit and Gradio for building web apps.

Step 4: Install and Configure Git

1. Install Git:
    - macOS often includes Git, but ensure it's up-to-date. In Terminal, type:

git --version

If Git is not installed or outdated, install it via Homebrew (a package manager for macOS):

- Install Homebrew if you don't have it:

/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"

- Then install Git:

brew install git

- Verify Git is installed:

git --version

You should see something like git version 2.45.2.

2. Configure Git with your details:
    - In Terminal, type:
    - git config --global user.name "Your Name"

git config --global user.email "your.email@example.com"

Replace Your Name with your name and your.email@example.com with the email tied to your GitHub account.

3. Log into GitHub:
    - Go to github.com in your browser.
    - Sign in with your username and password.
    - You'll need a 2FA code:
        - Open Microsoft Authenticator on your phone, find the GitHub code (a 6-digit number), and enter it.
4. Clone your project from GitHub:
    - In GitHub, go to your repository (e.g., your-username/ai-fellowship-test).
    - Click the green "Code" button, copy the HTTPS URL (e.g., https://github.com/your-username/ai-fellowship-test.git).
    - In Terminal (in /Users/YourUsername/ai/ai_fellowship), type:

git clone https://github.com/your-username/ai-fellowship-test.git

Press Enter. This downloads your files into a new folder ai-fellowship-test.

- Move the files to the main folder:
    - Open Finder, go to /Users/YourUsername/ai/ai_fellowship/ai-fellowship-test, select all files, drag them to /Users/YourUsername/ai/ai_fellowship.
    - Delete the empty ai-fellowship-test folder.

Step 5: Install GitHub Desktop

1. Go to desktop.github.com in your browser.
2. Click "Download for macOS" (e.g., GitHubDesktop.zip).
3. Open the downloaded file, drag GitHub Desktop to your Applications folder.
4. Launch GitHub Desktop from Applications or Spotlight.
5. Sign in to GitHub Desktop:

- Click “Sign in to GitHub.com.”
- Enter your username and password.
- Use Microsoft Authenticator to get your 2FA code and enter it.
- Authorize GitHub Desktop when prompted.

6. Add your project to GitHub Desktop:
    - In GitHub Desktop, go to File > Add Local Repository.
    - Choose /Users/YourUsername/ai/ai_fellowship and click “Add repository.”
    - You’ll see your project files in GitHub Desktop, ready to commit and push changes later.

Step 6: Install GitHub Copilot in VS Code

1. Open VS Code and go to the Extensions view (Cmd+Shift+X).
2. Search for “GitHub Copilot” and install the extension by GitHub.
3. Sign in to GitHub Copilot:
    - Follow the prompts in VS Code to sign in (it will open a browser).
    - Log in with your username, enter your 2FA code from Microsoft Authenticator, and authorize VS Code.
4. Test GitHub Copilot:
    - Open streamlit_test.py in VS Code (from /Users/YourUsername/ai/ai_fellowship).
    - Type a comment like # Function to say hello and press Enter.
    - Copilot should suggest code (e.g., def say_hello(): return "Hello!"). Press Tab to accept it.

Step 7: Configure VS Code to Run Streamlit and Gradio Apps

1. Open VS Code and open your project folder (/Users/YourUsername/ai/ai_fellowship via File > Open Folder).
2. Set the Python interpreter:
    - In the bottom-left corner, click the Python version (or “Select Interpreter”).

- Choose Python 3.13.4 ('venv': venv) (path: /Users/YourUsername/ai/ai_fellowship/venv/bin/python).

3. Create a launch.json file:
    - Go to the “Run and Debug” panel (Cmd+Shift+D).
    - Click the gear icon to create a launch.json file (select “Python” if prompted).
    - Replace the content with:
    - {
    - "version": "0.2.0",
    - "configurations": [
    - {
    - "name": "Streamlit: Run Current File",
    - "type": "python",
    - "request": "launch",
    - "module": "streamlit",
    - "args": ["run", "${file}"]
    - },
    - {
    - "name": "Python: Current File",
    - "type": "python",
    - "request": "launch",
    - "program": "${file}"
    - }
    -]

}

Step 8: Install Additional Tools

1. **Google Colab**:

- Go to colab.research.google.com.
- Sign in with your Google account (create one if needed at accounts.google.com).
- Click “New Notebook,” type print("Hello from Google Colab!") in the first cell, and click the “Run” button. You should see the output.

2. **Docker**:
    - Go to docker.com/products/docker-desktop.
    - Download “Docker Desktop for Mac” (select the version for Apple Silicon or Intel based on your Mac).
    - Open the downloaded file (e.g., Docker.dmg), drag Docker to your Applications folder.
    - Launch Docker Desktop from Applications.
    - Accept the agreement and verify it works:
        - In Terminal, type:

docker run hello-world

You should see “Hello from Docker!”

3. **Groq**:
    - Groq provides access to xAI’s Grok services (e.g., Grok 3, the AI assistant).
    - **Platform Access**:
        - Go to grok.com or x.com, log in with your account, and interact with Grok (e.g., ask questions).
    - **API Access** (if needed for your project):
        - Go to x.ai/api for instructions on API access.
        - Sign up for an API key if required (or check with fellowship coordinators for access details).
        - Follow the API documentation to test (e.g., using Python to make a request to Grok).

Step 9: Test Your Setup

- Open VS Code, load /Users/YourUsername/ai/ai_fellowship, and test your apps (Streamlit, Gradio) as in Step 7.
- Use GitHub Desktop to commit and push any new changes to your GitHub repository.
- Access Google Colab, Docker, and Groq as needed for your fellowship tasks.

Done!

You're all set to continue the **AI for Engineers Fellowship** on your Mac computer. If you need help, contact Tech Support (Shivaraj, hi@outskill.com).