National College of Ireland

AskFlow Configuration Manual

Configuration Manual
Master of Science in Artificial Intelligence

Shudong Wang
Student ID: x23115874

School of Computing
National College of Ireland

Supervisor: Arundev Vamadevan

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Shudong Wang
Student ID:	x23115874
Programme:	Master of Science in Artificial Intelligence
Year:	2025
Module:	Configuration Manual
Supervisor:	Arundev Vamadevan
Submission Due Date:	August 2025
Project Title:	AskFlow Configuration Manual
Word Count:	3500
Page Count:	11

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Shudong Wang
Date:	14th September 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☒
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☒
You must ensure that you retain a **HARD COPY** of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☒

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	Programme Coordinator
Date:	
Penalty Applied (if applicable):	

Declaration

I hereby certify that this material, which I now submit for assessment on the programme of study leading to the award of Configuration Manual, is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

I understand that plagiarism, collusion, and copying are grave and serious offences in the university and accept the penalties that would be imposed should I engage in plagiarism, collusion or copying.

I have read and understood the Assignment Regulations. I have identified and included the source of all facts, ideas, opinions and viewpoints of others in the assignment references. Direct quotations, paraphrasing, discussion of ideas and facts taken from books, journal articles, internet sources, module text, or any other source whatsoever are acknowledged and the sources cited are identified in the assignment references.

This assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other programme of study.

Name: Shudong Wang
Date: 14th September 2025

AskFlow Configuration Manual

Shudong Wang
x23115874

1 System Requirements

AskFlow deployment requires specific hardware and software configurations to ensure optimal performance. The system integrates local AI models, vector databases, and web services, necessitating adequate computational resources for stable operation.

1.1 Hardware Requirements

1.1.1 Memory Considerations

Memory allocation represents the primary performance constraint. AI models require substantial RAM allocation for optimal language processing capabilities:

- **Absolute minimum:** 8GB RAM - this will work, but expect slower performance
- **Recommended:** 16GB RAM - much more comfortable, especially when running multiple models
- **Optimal:** 32GB RAM - if you're planning to experiment with larger models or run multiple instances

The reason for these requirements is that modern language models like Llama2-7B need to be loaded entirely into memory. The 7-billion parameter model alone requires about 4-5GB of RAM, and that's before we account for the operating system, web browser, and other applications you might be running.

1.1.2 Storage Requirements

You'll need a decent amount of storage space, and the type of storage makes a real difference:

- **Minimum space:** 15GB free disk space
- **Recommended:** 25GB+ for comfortable operation and future model downloads
- **Storage type:** SSD strongly recommended - model loading from traditional hard drives can take several minutes

The storage breaks down roughly as follows: Ollama models (5-10GB), Python dependencies (1-2GB), application data and logs (1-3GB), plus some breathing room for temporary files during processing.

1.1.3 Processing Power

While not as critical as memory, your CPU does matter:

- **Architecture:** 64-bit processor (required)
- **Cores:** Multi-core processor recommended (4+ cores ideal)
- **GPU:** Optional but beneficial - NVIDIA GPUs with CUDA support can significantly speed up model inference

1.2 Operating System Support

AskFlow has been tested across the major operating systems, though some platforms are more straightforward than others:

1.2.1 Windows

- **Supported versions:** Windows 10 (build 1903+) or Windows 11
- **Notes:** Works well, though initial setup can be slightly more involved due to Python environment management
- **Considerations:** Windows Defender may flag some downloads - this is normal for AI model files

1.2.2 macOS

- **Supported versions:** macOS 10.15 (Catalina) or later
- **Notes:** Generally the smoothest installation experience
- **Apple Silicon:** M1/M2 Macs work excellently and often outperform Intel machines

1.2.3 Linux

- **Primary support:** Ubuntu 18.04 LTS and later
- **Other distributions:** Debian, CentOS, Fedora should work fine
- **Notes:** Often the best performance, especially on server deployments

1.3 Network Requirements

Don't overlook the network side of things:

- **Initial setup:** Reliable broadband connection for downloading models (several GB)
- **Ongoing operation:** Stable connection for Weaviate cloud database access
- **Bandwidth:** Not particularly demanding once set up, but initial model downloads can take 30+ minutes on slower connections

1.4 Software Prerequisites

Before diving into the installation, make sure you have these basics covered:

1.4.1 Python Environment

- **Version:** Python 3.8 or newer (3.9+ recommended)
- **Package manager:** pip should be included, but verify it's working
- **Virtual environments:** While not strictly required, strongly recommended for keeping things tidy

1.4.2 Development Tools

- **Git:** Essential for downloading the source code and managing updates
- **Text editor:** Any editor will do, but something with syntax highlighting helps
- **Terminal/Command prompt:** You'll be doing some command-line work

1.4.3 Administrative Access

Some installation steps require administrator privileges, particularly:

- Installing Ollama system-wide
- Installing Python packages globally (if not using virtual environments)
- Configuring firewall rules if needed

1.5 Performance Expectations

To set realistic expectations, here's what you can expect on different hardware configurations:

- **8GB RAM, HDD:** Functional but slow - first query may take 2-3 minutes, subsequent queries 30-60 seconds
- **16GB RAM, SSD:** Comfortable experience - first query under 30 seconds, subsequent queries 5-15 seconds
- **32GB RAM, SSD, GPU:** Excellent performance - first query under 15 seconds, subsequent queries 2-5 seconds

These times include model loading, document processing, and response generation. Your mileage may vary depending on query complexity and document size.

2 System Installation

This section provides comprehensive installation procedures for all AskFlow system components. Follow these steps sequentially to ensure proper system deployment.

2.1 Prerequisites

System Requirements:

- Operating System: macOS 10.15+, Ubuntu 18.04+, or Windows 10+
- Memory: Minimum 8GB RAM (16GB recommended)
- Storage: 10GB available disk space
- Network: Internet connection for model downloads
- Python: Version 3.8 or higher

2.2 Step 1: Ollama Installation and Configuration

Ollama provides the local AI model runtime environment for AskFlow processing.

2.2.1 Installation Process

macOS and Linux Systems:

1. Open terminal application
2. Execute the installation command:

   ```
   curl -fsSL https://ollama.ai/install.sh | sh
   ```

3. Verify installation completion by checking version:

   ```
   ollama --version
   ```

Windows Systems:

1. Navigate to `https://ollama.ai/download`
2. Download the Windows installer executable
3. Right-click installer and select "Run as administrator"
4. Complete installation wizard with default settings
5. Verify installation through Command Prompt:

   ```
   ollama --version
   ```

2.2.2 Required Model Installation

Install the necessary AI models for system operation:

```
# Install embedding model for document vectorization
ollama pull nomic-embed-text

# Install primary language model (7B parameters)
ollama pull llama2:7b

# Verify model installation
ollama list
```

Expected Output:

```
NAME                    ID              SIZE    MODIFIED
llama2:7b               e38ae474bf8c    3.8GB   2 hours ago
nomic-embed-text        0a109f422b47    274MB   2 hours ago
```

2.3 Step 2: Weaviate Database Setup

Weaviate provides vector database functionality for document embedding storage and similarity search operations.

2.3.1 Cloud Instance Creation

1. Access Weaviate Cloud Console at `https://console.weaviate.cloud/`
2. Create new account using institutional email address
3. Verify email address through confirmation link
4. Log in to console dashboard
5. Create new cluster with following specifications:
   - Cluster name: `askflow-production`
   - Region: Select nearest geographical location
   - Tier: Sandbox (free tier sufficient for evaluation)
6. Record cluster URL and API key for configuration

2.4 Step 3: Python Environment Setup

2.4.1 Virtual Environment Creation

1. Create project directory:

```
mkdir askflow-system
cd askflow-system
```

2. Create Python virtual environment:

   ```
   python -m venv askflow-env
   ```

3. Activate virtual environment:
   macOS/Linux:

   ```
   source askflow-env/bin/activate
   ```

   Windows:

   ```
   askflow-env\Scripts\activate
   ```

2.4.2 Dependency Installation

Install required Python packages:

```
# Upgrade pip to latest version
pip install --upgrade pip

# Install core dependencies
pip install fastapi uvicorn weaviate-client
pip install openai anthropic google-generativeai cohere
pip install python-multipart python-jose[cryptography]
pip install beautifulsoup4 requests pandas numpy

# Install development dependencies
pip install pytest pytest-asyncio black flake8
```

3 System Configuration

This section details the configuration procedures, environment setup, and service initialization required for AskFlow system operation.

3.1 Step 4: Environment Configuration

3.1.1 API Key Configuration

Create environment configuration file for API credentials:

1. Create configuration file in project root:

   ```
   touch .env
   ```

2. Add required API keys to `.env` file:

```
# Weaviate Configuration
WEAVIATE_URL=https://your-cluster.weaviate.network
WEAVIATE_API_KEY=your_weaviate_api_key

# AI Provider API Keys
OPENAI_API_KEY=your_openai_api_key
ANTHROPIC_API_KEY=your_anthropic_api_key
GOOGLE_API_KEY=your_google_api_key
COHERE_API_KEY=your_cohere_api_key

# Application Configuration
JWT_SECRET_KEY=your_jwt_secret_key
REDIS_URL=redis://localhost:6379
LOG_LEVEL=INFO
```

3. Set appropriate file permissions:

```
chmod 600 .env
```

3.1.2 Application Configuration

Create main configuration file `config.yaml`:

```
# AskFlow System Configuration
server:
  host: "0.0.0.0"
  port: 8000
  workers: 4

database:
  weaviate:
    class_name: "Documents"
    vectorizer: "text2vec-openai"

ai_providers:
  routing_strategy: "cost_optimized"
  fallback_provider: "ollama"

  providers:
    openai:
      model: "gpt-3.5-turbo"
      max_tokens: 1000
      temperature: 0.1

    anthropic:
      model: "claude-3-sonnet-20240229"
      max_tokens: 1000
      temperature: 0.1
```

```
    ollama:
      model: "llama2:7b"
      base_url: "http://localhost:11434"

logging:
  level: "INFO"
  format: "%(asctime)s - %(name)s - %(levelname)s - %(message)s"
  file: "askflow.log"
```

3.2 Step 5: Service Initialization

3.2.1 Ollama Service Startup

Initialize Ollama service before application launch:

1. Start Ollama service daemon:

   ```
   ollama serve
   ```

2. Verify service availability:

   ```
   curl -s http://localhost:11434/api/tags | jq .
   ```

   Expected Response:

   ```
   {
     "models": [
       {
         "name": "llama2:7b",
         "modified_at": "2024-01-15T10:30:00Z",
         "size": 3826793677
       },
       {
         "name": "nomic-embed-text",
         "modified_at": "2024-01-15T10:25:00Z",
         "size": 274301440
       }
     ]
   }
   ```

3.2.2 Application Launch Procedure

Execute the following sequence to start AskFlow system:

1. Activate Python virtual environment:

```
source askflow-env/bin/activate
```

2. Initialize database schema:

```
python scripts/init_database.py
```

3. Start application server:

```
uvicorn main:app --host 0.0.0.0 --port 8000 --reload
```

Expected Startup Log:

```
INFO:     Uvicorn running on http://0.0.0.0:8000 (Press CTRL+C to quit)
INFO:     Started reloader process [12345] using StatReload
INFO:     Started server process [12346]
INFO:     Waiting for application startup.
INFO:     Weaviate connection established
INFO:     AI providers initialized: OpenAI, Anthropic, Ollama
INFO:     Application startup complete.
```

4 Troubleshooting

4.1 Common Issues

4.1.1 Ollama Service Not Starting

Problem: Connection refused to `http://localhost:11434`
Solution:

1. Check if service is running:

```
ps aux | grep ollama
```

2. Restart service:

```
pkill ollama
ollama serve
```

3. Verify models are loaded:

```
ollama list
```

4.1.2 Application Won't Start

Problem: Python environment or dependency issues
Solution:

1. Recreate virtual environment:

```
rm -rf askflow-env
python -m venv askflow-env
source askflow-env/bin/activate
pip install -r requirements.txt
```

2. Check environment variables:

```
echo $WEAVIATE_URL
echo $WEAVIATE_API_KEY
```

4.1.3 Slow Query Performance

Problem: Queries taking more than 10 seconds
Solution:

1. Reduce model context:

```
export OLLAMA_MAX_CONTEXT=2048
```

2. Limit concurrent requests:

```
export UVICORN_WORKERS=2
```

3. Monitor system resources:

```
htop
```

5 Maintenance

This section covers ongoing maintenance tasks and system optimization procedures.

5.1 Regular Maintenance Tasks

5.1.1 Weekly Tasks

- Monitor system resource usage
- Check application logs for errors
- Verify service health status

5.1.2 Monthly Tasks

- Update AI models if new versions available
- Review and rotate API keys
- Clean up temporary files and logs

5.2 Performance Optimization

5.2.1 Memory Management

For optimal performance:

- Monitor memory usage during operation
- Close unnecessary applications
- Consider using smaller models for resource-constrained systems

5.2.2 Storage Optimization

- Use SSD storage for faster model loading
- Ensure adequate free space (15-20GB recommended)
- Regular cleanup of temporary files