

# Infrastructure Stop & Start Operational Procedures

Project Deliverable

---

**Course** B1AI

**Class:**

**Author:** François LANGE

**Date:** 2026/06/19



# Table of Contents

- • Sequence Priority Rules ..... Page 2
- • Startup Procedures ..... Page 3
  - Step 2.1: Start the Core MySQL Database ..... Page 3
- Verify database logs ..... Page 3
  - Step 2.2: Start AI Engine & SearXNG Search ..... Page 3
- Check that Ollama responds ..... Page 3
  - Step 2.3: Start Streamlit App and Nginx Gateway ..... Page 3
- Verify Web Interface status ..... Page 3
  - Step 2.4: Start Zabbix Monitoring Suite ..... Page 3
- Check dashboard availability ..... Page 3
  - • Shutdown Procedures ..... Page 3
  - • Stop Monitoring Components ..... Page 3
  - • Stop Web Frontend and Proxy Gateway ..... Page 3
  - • Stop NLP and Search Services ..... Page 3
  - • Stop Database Container gracefully ..... Page 3
    - • Status Verification Commands ..... Page 2
- List all running containers in the stack ..... Page 4
- Inspect raw container logs for error spikes ..... Page 4
- Verify TCP socket listener binds ..... Page 4

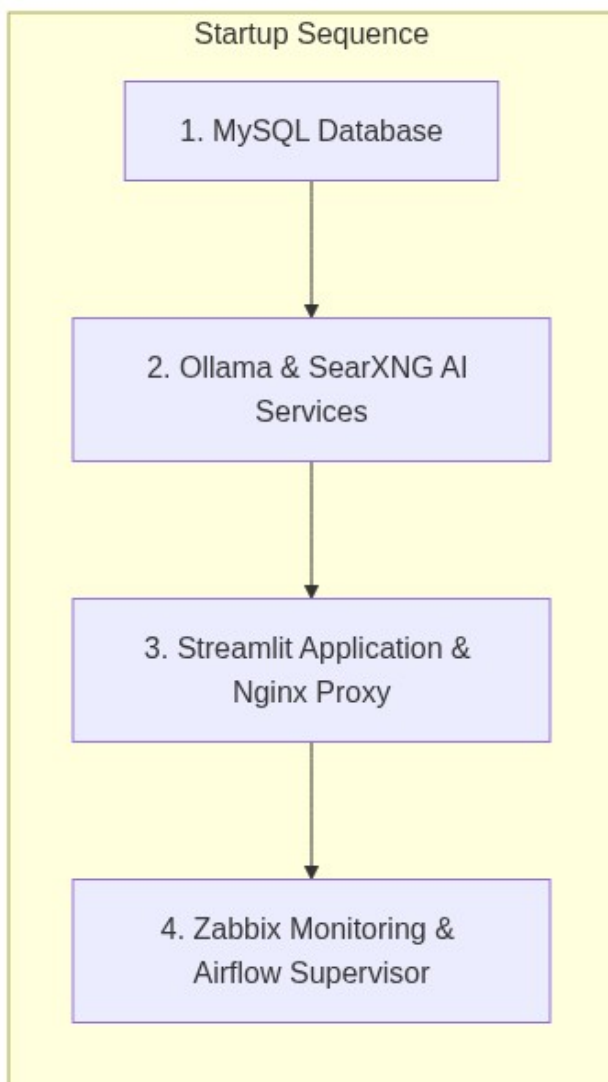
The current version is #ident  
 "@(#)\$Format:LocalFoodAI\_lanfr144:generate\_docs.py:Francois  
 Lange:lanfr144@school.lu:2026/06/16 21:48:22:Francois  
 Lange:lanfr144@school.lu:2026/06/16  
 21:48:22:2a8ed056889f3b796f9266feda591b12b72f3b96:HEAD -> main, origin/main:\$"

# Infrastructure Stop & Start Operational Procedures

This runbook outlines the exact sequence and commands to start, stop, and verify each microservice in the Local Food AI environment.

## 1. Sequence Priority Rules

Due to database socket requirements and network bindings, services **must** be started and stopped in the following order:



## 2. Startup Procedures

### Step 2.1: Start the Core MySQL Database

Verify that the database service is up and listening on port 3307:

```
docker compose up -d mysql
# Verify database logs
docker compose logs -f mysql
```

### Step 2.2: Start AI Engine & SearXNG Search

Deploy the AI components:

```
docker compose up -d ollama searxng
# Check that Ollama responds
curl http://localhost:11434/api/tags
```

### Step 2.3: Start Streamlit App and Nginx Gateway

Bring up the frontend web interface and reverse proxy:

```
docker compose up -d app nginx
# Verify Web Interface status
curl -I http://localhost
```

### Step 2.4: Start Zabbix Monitoring Suite

Deploy the monitoring server and agents:

```
docker compose up -d zabbix-server zabbix-web zabbix-agent
# Check dashboard availability
curl -I http://localhost:8081
```

## 3. Shutdown Procedures

To perform system maintenance or schema migration, stop services in reverse order to prevent lockups:

```
# 1. Stop Monitoring Components
docker compose stop zabbix-agent zabbix-web zabbix-server

# 2. Stop Web Frontend and Proxy Gateway
docker compose stop nginx app

# 3. Stop NLP and Search Services
docker compose stop searxng ollama

# 4. Stop Database Container gracefully
```

```
docker compose stop mysql
```

## 4. Status Verification Commands

Use these commands to verify container state and port bindings:

```
# List all running containers in the stack
docker compose ps

# Inspect raw container logs for error spikes
docker compose logs --tail=100

# Verify TCP socket listener binds
netstat -tulpn | grep -E "80|3307|8081|11434"
```

## References

- **OpenFoodFacts Dataset & API Catalog:** Detailed food ingredients database. (<https://world.openfoodfacts.org/>)
- **Ollama Local LLM Inference Engine:** Lightweight instruction-following llama3.2 runtimes. (<https://ollama.com/>)
- **Zabbix Enterprise Telemetry and Monitoring:** System health and performance logging. (<https://www.zabbix.com/>)

## Index

- **AI:** Page 1, Page 2, Page 3, Page 4, Page 5
- **MySQL:** Page 1, Page 3, Page 4, Page 5
- **Zabbix:** Page 1, Page 3, Page 4, Page 5
- **Docker:** Page 3, Page 4, Page 5
- **Streamlit:** Page 1, Page 3, Page 5
- **Nginx:** Page 1, Page 3, Page 5
- **RAG:** Page 5
- **Allergens:** Page 5
- **Vitamins:** Page 5
- **Minerals:** Page 5
- **Clinical:** Page 5
- **WSL:** Page 5
- **Ollama:** Page 1, Page 3, Page 4, Page 5
- **LLM:** Page 4, Page 5
- **Database:** Page 1, Page 2, Page 3, Page 4, Page 5
- **Security:** Page 5
- **Telemetry:** Page 4, Page 5
- **Backup:** Page 5
- **Firewall:** Page 5
- **SMTP:** Page 5