

---

  

---

  

```
flowchart TD
  subgraph "Client Layer"
    Browser["Clinician Browser"]
  end

  subgraph "Gateway & Application Nodes"
    Nginx["Nginx Reverse Proxy  
(Port 80)"]
    Streamlit["Streamlit Web App  
(Port 8502 / Docker Container)"]
  end
```

```
end

subgraph "Intelligence & Search Nodes"
  Ollama["Ollama Daemon
(Port 11434 / Docker Container)"]
  SearXNG["SearXNG Meta-Search
(Port 8085 / Docker Container)"]
end

subgraph "Data Storage & Observability Nodes"
  MySQL["MySQL Database Server
(Port 3306 / Docker Container)"]
  Zabbix["Zabbix Server & Agent
(Ports 10051 & 10050)"]
  ZabbixWeb["Zabbix Web Dashboard
(Port 8081)"]
end

%% Communication paths
Browser -->|HTTP| Nginx
Nginx -->|Reverse Proxy Pass| Streamlit
Streamlit -->|EAV & FULLTEXT SQL queries| MySQL
Streamlit -->|Local Chat Inference / RAG| Ollama
Streamlit -->|Tool-Calling search queries| SearXNG
Streamlit -->|SNMP Traps / Telemetry| Zabbix
ZabbixWeb -->|Queries metrics| Zabbix
```

---

.env

```
NETWORK_MODE=server
LLM_MODEL=llama3.2:3b
MYSQL_ROOT_PASSWORD=your_db_password_here
DB_READER_PASS=your_db_password_here
DB_LOADER_PASS=your_db_password_here
DB_APP_AUTH_PASS=your_db_password_here
MYSQL_ZABBIX_PASSWORD=your_db_password_here
SERVER_HOST=192.168.130.170
SERVER_USER=francois
SERVER_PASS=your_db_password_here
```

app

.env

```
app:
  build:
    context: .
    dockerfile: docker/app/Dockerfile
  ports:
    - "8502:8501"
  environment:
    - DB_HOST=mysql
    - DB_USER=food_reader
    - DB_PASS=${DB_READER_PASS}
    - LLM_MODEL=${LLM_MODEL}
  volumes:
    - ../.env:/app/.env
```

```
docker compose up -d --build
```

```
docker compose -f docker-compose_skip.yml up -d --build
```

```
chmod +x manage_services.sh
./manage_services.sh restart
```

---

app.py

docs/

\$Format:

```
read_file write_file c:
\Users\lanfr144\Documents\DOPR01\Antigravity\Food
/tmp .gemini
command git python chmod docker-compose Get-
Content
read_url execute_url
192.168.130.170 192.168.130.161
```

---

```
git-ident-filter.py
.*?[^$]*?$
$Format: app.py
```

SyntaxError: unterminated string literal

), ensuring it never matches across newline boundaries. 2. We split the string literal searches inside so they are physically split across concatenated strings (e.g.

```
local_tools/git-ident-filter.py
```

```
git checkout HEAD -- local_tools/git-ident-filter.py
```

errors='replace'

---

llama3.2:3b

food-ollama-1

---

/var/log/nginx/access.log

192.168.1.50

127.0.0.1

mysql app

docker-compose.yml

app

8502

tcpdump

tcpdump -i eth0 dst port not 80 and dst port not 22 and dst port no