

\$Id\$

Final Project Report (Living Document)

What Has Been Done

1. Core Architecture: Deployed a resilient 8-container local fallback Docker Compose stack (MySQL, Streamlit UI, local Ollama LLM, anonymous SearXNG search, secure Nginx proxy, and local Zabbix Server/Web/Agent observability suite).
2. Database Optimization: Successfully loaded OpenFoodFacts records and utilized advanced vertical partitioning and FULLTEXT indices.
3. Clinical Subquery Strategy: Refactored the core Pandas/SQL query pipeline to use subquery limiting, resolving Cartesian join explosions and reducing query latency to ~0.04s.
4. Monitoring & Security: Nginx securely proxies traffic on Port 80. Zabbix actively monitors proxy and server health, dynamically handling SNMP/alert loops in local/offline fallback mode.
5. Git Versioning: Implemented `Git .gitattributes` to push \$Id\$ tracking directly into the Python Application UI.

What Needs To Be Done (Day 2 Operations)

1. SSL/TLS Certificates: The Nginx proxy is functional on HTTP port 80. Port 443 (HTTPS) must be configured with a Let's Encrypt certificate for true production encryption.
2. User Acceptance Testing (UAT): Clinical dietitians should rigorously test the AI Chat constraints and Plate Builder to ensure edge cases are handled safely.
3. Advanced Rate Limiting: Limit the number of AI requests per user using a sliding window algorithm in `app.py`.

What Is The Next Step

- Execute the `data_sync.sh` cron job monthly.
- Maintain the automated `backup_db.sh` 7-day retention cycle.
- Begin the hand-off to the operational team for Phase 2 feature requests.