

Task 7-Deployment: Logging & Monitoring Report

Prepared by: Aymane Salhi

Date: July 27, 2025

1. Task Objective

Integrate request/response logging and Prometheus-based monitoring into the existing inference API, ensuring the team can deploy and observe the service via Docker Compose.

2. Challenges Encountered

- The provided `ml_inference_api.rar` contained over 7,000 files (historical backups), making extraction and integration impractical.
- To expedite delivery, I focused exclusively on the **Task7-Deployment** folder (a self-contained ZIP), as it contained the live API code.

3. Solution & Implementation

3.1 Logging Integration

- Created **logger1.py**: configured Python's built-in logging to write INFO+ messages to both `api.log` and console.
- Modified **api_app.py**:
 - Imported logger from `logger1.py`.
 - Added an HTTP middleware function to log each incoming request and outgoing response (including methods, paths, status codes, and errors).

3.2 Monitoring Integration

- Installed the **prometheus-client** library dependency.

- Extended **api_app.py**:
 - Imported Prometheus Counter, Histogram, and related functions.
 - Defined two metrics: request count (req_count) and request latency (req_latency_sec).
 - Enhanced middleware to increment counters and observe latency for each request.
 - Exposed a new endpoint **GET /metrics** to serve Prometheus-formatted metrics.

3.3 Configuration & Packaging

- Added **prometheus.yml**: configures Prometheus to scrape localhost:8000/metrics every 15 seconds.
- Added **docker-compose.yml**: defines three services—app (FastAPI), prometheus, and grafana—allowing one-command startup.
- Updated **requirements.txt** to include prometheus-client.

4. Files Created or Modified

File Name	Action	Description
logger1.py	Created	Central logging configuration
api_app.py	Modified	Added logging & Prometheus middleware + /metrics
prometheus.yml	Created	Prometheus scrape configuration
docker-compose.yml	Created	Docker Compose stack for app, Prometheus, Grafana
requirements.txt	Updated	Added prometheus-client dependency

5. Package Details

- The updated **Task7-Deployment** folder has been compressed to **Task7-Deployment_logging.zip**.
- Location: **Final Task → Logging & Monitoring – SALHI Aymane** on the team drive.

6. Next Steps / Recommendations

1. **Review & merge** the changes into the main repository.
2. **Deploy** the stack via `docker-compose up -d` and verify:
 - a. Service health: `GET /health`
 - b. Logging output: `tail api.log` or console logs
 - c. Metrics endpoint: `GET /metrics`
 - d. Grafana dashboard can be configured to visualize metrics.
3. (Optional) Integrate similar changes into the full `ml_inference_api` archive once a cleaned version is available.

End of Report