

Raport Wydajności Systemu Primus 2026

1. Środowisko Testowe

Testy zostały przeprowadzone na lokalnym środowisku deweloperskim (Docker).

Specyfikacja Sprzętowa:

- CPU:** Intel Core i7-13700KF
- GPU:** NVIDIA GeForce RTX 3060 Ti (wykorzystywane przez Ollama/Local LLM)
- RAM:** 32 GB

Konfiguracja Programowa:

- Backend:** FastAPI + Uvicorn (4 workery)
- Baza Danych:** PostgreSQL 15 (AsyncPG, Isolation Level: Read Committed)
- AI Engine:** Ollama (Model: qwen3:4b)
- Narzędzie Testowe:** Locust (Headless Mode)

2. Scenariusze Testowe ("Extreme")

A. Scenariusz Standardowy (200 Użytkowników)

Symulacja intensywnej pracy magazynu: 200 jednoczesnych pracowników wykonujących operacje wyszukiwania, alokacji i generowania raportów.

Komenda: poetry run locust -f scripts/locustfile.py StandardUser -u 200 -r 10 --run-time 1m

Wyniki:

Metryka	Wartość	Komentarz
Liczba Żądań (Total)	8,197	W ciągu 60 sekund
Przepustowość (RPS)	~136 req/s	Stabilne obciążenie
Opóźnienie (Odczyt)	9 ms (mediana)	Wyszukiwanie produktów, listowanie stanów
Opóźnienie (Zapis)	~500 ms (mediana)	Transakcyjne operacje magazynowe
Opóźnienie (Raport)	120 ms (mediana)	Generowanie PDF (Celery async)
Błędy	0%	System w pełni stabilny

B. Scenariusz AI (20 Użytkowników)

Symulacja 20 pracowników jednocześnie wydających komendy głosowe. Jest to test skrajny dla pojedynczej karty graficznej.

Komenda: poetry run locust -f scripts/locustfile.py AIUser -u 20 -r 2 --run-time 1m

Wyniki:

Metryka	Wartość	Komentarz
Liczba Żądań (Total)	276	Limitowane przez prędkość inferencji GPU
Przepustowość (RPS)	~4.2 req/s	Kolejkowanie zadań w Ollama
Opóźnienie (Mediana)	3,500 ms	Czas oczekiwania w kolejce + inferencja
Opóźnienie (Max)	5,800 ms	I tak poniżej timeoutu (120s)
Błędy	0%	Brak odrzuconych połączeń (Connection Refused/Timeout)

3. Wnioski

- Backend (FastAPI/Postgres):** Jest ekstremalnie wydajny na procesorze i7-13700KF. Obsługa 200 jednoczesnych użytkowników (co odpowiada magazynowi zatrudniającemu ok. 1000 osób na zmianę) skutkuje czasami odpowiedzi rzędu milisekund.
- AI (Ollama/GPU):** Karta RTX 3060 Ti radzi sobie z kolejkowaniem 20 równoległych zapytań. Opóźnienie rośnie liniowo, ale system pozostaje stabilny. Dla produkcji zalecane jest skalowanie horyzontalne (więcej instancji Ollama) lub użycie komercyjnego API (OpenAI) przy tak dużym obciążeniu.
- Wniosek:** Są to wypełni wystarczające wyniki dla większości zastosowań, a otrzymane zostały na tanim sprzęcie konsumenckim.