

Instrukcja Uruchomienia Serwera — System Primus 2026

To repozytorium odpowiada za orkiestrację usług, zarządzanie siecią i bezpieczeństwo danych (TLS/Szyfrowanie).

Konfiguracja Środowiska

1. Tryb Produkcyjny (Zalecany)

W tym trybie pobierasz tylko repozytorium infrastruktury i uruchamiasz gotowe obrazy z Docker Hub.

2. Tryb Deweloperski (Lokalne Budowanie)

Wymaga pobrania wszystkich repozytoriów, aby zbudować obrazy z kodu źródłowego.

Struktura Katalogów

Utwórz katalog główny i wejdź do niego:

Klonowanie Repozytoriów

Wykonaj poniższe polecenia, aby pobrać wszystkie moduły systemu:

Uruchomienie

Flaga `--build` wymusi zbudowanie obrazów z kodu źródłowego (`./primus-backend` itd.).

Struktura Projektu

- **dev/** [DOMYSLNE]: Zawiera konfigurację deweloperską (buduje obrazy lokalnie z `context: .`).

- **`docker-compose.yml`**: Konfiguracja produkcyjna (używa gotowych obrazów z Docker Hub).
- **`docker-compose-nvidia.yml`**: Dodatek produkcyjny dla NVIDIA GPU.
- **`docker-compose-amd.yml`**: Dodatek produkcyjny dla AMD GPU.

Konfiguracja NVIDIA (GPU)

Zalecana dla systemów z kartami graficznymi NVIDIA.

Instalacja NVIDIA Container Toolkit:

Uruchomienie:

Konfiguracja AMD (GPU)

Tryb Hybrydowy (WSL2)

1. **Ollama Windows:** `$env:OLLAMA_VULKAN="1"; ollama serve`
2. **Konfiguracja `.env`:** `OLLAMA_URL=http://host.docker.internal:11434/api/generate`
3. **Uruchomienie:** `docker compose up --build -d`

Tryb Natywny (Linux ROCm)

Konfiguracja z zewnętrznym Ollama/OpenAI (CPU)

Zalecana dla systemów bez dedykowanej karty graficznej, z niekompatybilną kartą graficzną lub do szybkich testów.

Uruchomienie:

[!IMPORTANT] W pliku `.env` należy ustawić wszystkie wymagane zmienne, w tym adresy serwerów i ewentualne klucze API.

Certyfikaty SSL

System generuje certyfikaty self-signed przy pierwszym uruchomieniu (`scripts/generate_certs.sh`).

Własne Certyfikaty

Podmień pliki w `nginx/certs/`: `nginx.crt`, `nginx.key`, `rootCA.pem`. Następnie:
`docker compose restart nginx redis mosquitto`

Dokumentacja Projektu

Pełna dokumentacja architektury, modelu danych i modułów znajduje się w osobnym repozytorium: [primus-docs](#)