

NipService WebAPI wersja 1.0.0

Aplikacja do walidacji numerów NIP/VAT (PL i UE)

Spis treści

1. [Ogólne informacje](#)
 2. [Wymagania](#)
 3. [Instalacja](#)
 4. [Użytkowanie](#)
 5. [Wersje aplikacji](#)
-

1. Ogólne informacje

NipService jest usługą WebAPI do walidacji numerów NIP/VAT:

- Walidacja polskich NIP (Biała Lista MF)
 - Walidacja VAT UE (system VIES)
 - Healthcheck `/api/health` i endpoint wersji `/api/Version`
 - Logowanie zdarzeń za pomocą NLog
-

2. Wymagania

2.1. Wymagania sprzętowe

- Procesor min. 2 rdzenie
- 2 GB pamięci RAM
- 500 MB przestrzeni dyskowej

2.2. Wymagania oprogramowania

- ASP.NET Core Runtime 10.0
- Serwer WWW (opcjonalnie):
 - Windows: IIS lub Apache
 - Linux: Nginx (rekomendowany) lub Apache
- Docker/Podman (opcjonalnie, konteneryzacja)

2.3. Wymagania dodatkowe (sieć)

- Dostęp do zewnętrznych usług:
 - Biała Lista MF: <https://wl-api.mf.gov.pl>
 - VIES: https://ec.europa.eu/taxation_customs/vies/rest-api

Sprawdzenie dostępności adresów

Windows (PowerShell):

```
Test-NetConnection w1-api.mf.gov.pl -Port 443
Test-NetConnection ec.europa.eu -Port 443
```

Linux:

```
curl -f -s -o /dev/null -w "%{http_code}\n" https://w1-api.mf.gov.pl
curl -f -s -o /dev/null -w "%{http_code}\n" https://ec.europa.eu
```

3. Instalacja

3.1. NipService downloads

W przypadku hostowania we własnej infrastrukturze aplikacja dostępna jest wersja:

- **Produkcyjna:** 1.0.0 – dostępna [tutaj](#)

Najnowsza dokumentacja jest dostępna [tutaj](#). Dokumentacja w języku angielskim jest dostępna [tutaj](#).

3.2. Instalacja na systemach Linux

Instalacja tradycyjna (z serwerem WWW)

Aby uruchomić aplikację w systemie Linux należy zainstalować i skonfigurować serwer WWW. Rekomendowane serwery WWW to:

- **Nginx** – instrukcja instalacji dostępna [tutaj](#)
- **Apache** – instrukcja instalacji dostępna [tutaj](#)

Dodatkowo należy zainstalować środowisko uruchomieniowe ASP.NET Core w wersji 10. Instrukcja jak zainstalować i skonfigurować ASP.NET Core Runtime dla systemów Linux znajduje się [tutaj](#).

Po zainstalowaniu i konfiguracji serwera WWW należy rozpakować otrzymaną paczkę `NipService.WebAPI.zip` do katalogu głównego serwera WWW.

Instalacja z Docker

Wymagania:

- Docker Engine 20.10+
- Docker Compose (opcjonalnie)

Kroki instalacji:

1. Pobierz kod aplikacji lub obraz Docker
2. Utwórz katalog `logs`
3. Utwórz katalog `data-protection-keys`
4. Uruchom kontener:

```
# Uruchomienie kontenera (werjsa produkcyjna)
docker run -d -p 5000:8080 -v ./logs:/app/logs -v ./data-protection-keys:/app/data-protection-keys -
```

Lub z Docker Compose:

Przykładowy plik docker-compose.yml można pobrać [tutaj](#)

```
# Uruchomienie
docker-compose up -d

# Sprawdzenie statusu
docker-compose ps

# Wyświetlanie logów
docker-compose logs -f nipService-api

# Zatrzymanie
docker-compose down
```

Sprawdzenie działania:

```
curl http://localhost:5001/api/health
```

Aby przetestować czy aplikacja została zainstalowana prawidłowo można wywołać punkt końcowy **Version** (patrz sekcja 4.8).

Dokładna instrukcja instalacji z użyciem kontenerów dostępna [tutaj](#).

3.3. Instalacja na systemach Windows

W systemie Windows należy skonfigurować serwer WWW. Rekomendowanym serwerem dla systemów Windows jest **IIS**. Możliwe jest też użycie serwera Apache – instrukcja instalacji i konfiguracji dostępna [tutaj](#).

Dodatkowo należy zainstalować **ASP.NET Core Hosting Bundle** w wersji 10. Instrukcja instalacji .NET Hosting Bundle w wersji 10 znajduje się [tutaj](#).

WAŻNE: Należy zainstalować Hosting Bundle (nie samo Runtime), który zawiera zarówno ASP.NET Core Runtime jak i moduł IIS.

Po zainstalowaniu Hosting Bundle, w konfiguracji **Application Pool** w IIS należy ustawić **.NET CLR Version** na **“No Managed Code”**.

Po zainstalowaniu i konfiguracji serwera WWW należy rozpakować otrzymaną paczkę `NipService.WebAPI.zip` do katalogu głównego serwera WWW.

Aby przetestować czy aplikacja została zainstalowana prawidłowo można wywołać punkt końcowy **Version** (patrz sekcja 4.8).

3.4. Konfiguracja proxy

W pliku appsettings.json znajduje się sekcja odpowiedzialna za konfigurację proxy, którą należy uzupełnić aby aplikacja Nip Service mogła komunikować się z zewnętrznymi serwisami poprzez proxy. Należy zmienić właściwość "Enabled" na true i w polach Address, Port, Username i Password wprowadzić informacje o proxy (username i password są opcjonalne i powinny być tylko wypełnione w przypadku gdy proxy wymaga uwierzytelnienia)

```
"Validation": {
  "Proxy": {
    "Enabled": false,
    "Address": "http://proxy.example.com",
    "Port": 8080,
    "Username": null,
    "Password": null,
    "BypassOnLocal": true,
    "BypassList": []
  }
},
```

4. Użytkowanie

4.1. Walidacja NIP (PL) – Biała Lista MF

Endpoint: GET /api/NipCheck/{nip}

Opis: Walidacja podstawowa NIP (10 cyfr).

Odpowiedź: ValidationStatusWithNameResponse

```
{
  "isNipActive": true,
  "fastVerLast": "2025-01-01T12:00:00Z",
  "response": "OK",
  "name": "Firma Sp. z o.o.",
  "statusVat": "Czynny",
  "requestDate": "2025-01-01T12:00:00Z",
  "address": "Przykładowa 1, 00-000 Warszawa"
}
```

Endpoint: GET /api/NipCheck/{nip}/{date}

Opis: Walidacja NIP na konkretny dzień (format daty: RRRR-MM-DD).

Odpowiedź: ValidationStatusWithNameResponse (jak wyżej).

4.2. Walidacja VAT UE (VIES)

Endpoint: GET /api/ViesNipCheck/{nip}/{country}

Opis: Podstawowa walidacja numeru VAT w danym kraju (kod kraju: np. PL, DE).

Odpowiedź: ValidationStatusResponse

```
{
  "isNipActive": true,
  "fastVerLast": "2025-01-01T12:00:00Z",
```

```
"response": "OK",
"requestDate": "2025-01-01T12:00:00Z"
}
```

Endpoint: GET /api/ViesNipCheck/{nip}/{country}/{reqNip}/{reqCountry}

Opis: Walidacja z danymi zapytującego (requester).

Odpowiedź: ValidationStatusResponse (jak wyżej).

Endpoint (rozszerzony): GET /api/ViesNipCheck/extended/{nip}/{country}/{reqNip}/{reqCountry}

Opis: Walidacja rozszerzona – zwraca nazwę i status VAT.

Odpowiedź: ValidationStatusWithNameResponse

```
{
  "isNipActive": true,
  "fastVerLast": "2025-01-01T12:00:00Z",
  "response": "OK",
  "name": "Firma Sp. z o.o.",
  "statusVat": "Czynny",
  "requestDate": "2025-01-01T12:00:00Z",
  "address": "Przykładowa 1, 00-000 Warszawa"
}
```

4.3. Healthcheck

Endpoint: GET /api/Health

Opis: Zwraca status aplikacji oraz statusy usług zewnętrznych.

Odpowiedź: HealthResponse

Kody: 200 OK (healthy) / 503 Service Unavailable (degraded/unhealthy).

4.4. Wersja aplikacji

Endpoint: GET /api/Version

Odpowiedź: VersionResponse

```
{ "versionId": "1.0.0", "buildDate": "2025-01-01T12:00:00Z" }
```

5. Wersje aplikacji

Historia wersji

Wersja 1.0.0 (aktualna)

Lista funkcjonalności:

- Walidacja NIP (PL) – Biała Lista MF:
 - GET /api/NipCheck/{nip}
 - GET /api/NipCheck/{nip}/{date}

- Walidacja VAT UE (VIES):
 - `GET /api/ViesNipCheck/{nip}/{country}`
 - `GET /api/ViesNipCheck/{nip}/{country}/{reqNip}/{reqCountry}`
 - `GET /api/ViesNipCheck/extended/{nip}/{country}/{reqNip}/{reqCountry}`
 - Healthcheck aplikacji i usług zewnętrznych:
 - `GET /api/Health`
 - Informacje o wersji:
 - `GET /api/Version`
 - Dokumentacja Swagger UI dostępna pod `/swagger` (jeśli włączona w konfiguracji).
-

Data ostatniej aktualizacji dokumentacji: 2025-01-29

Wersja dokumentacji: 1.0

Zgodność z wersją aplikacji: 1.0.0+