DripTronic

automatyczne smarowanie łańcucha · OBD · tripy

Home Assistant — krok po kroku

Podgląd pozycji i danych olejarki w twojej instalacji

- Szybki start: sześć kroków, około dziesięciu minut
- Bez integracji, bez HACS, bez GitHuba — tylko jeden plik
- Gotowy plik YAML do pobrania razem z tą instrukcją
- Co zrobić, gdy dane nie przychodzą

0.6.146 · 159

Dokument dla wersji aplikacji 0.6.146 (versionCode 159).
Wydanie: październik 2026 · język: polski

by rumunn95

Spis treści

1 Szybki start — sześć kroków 3
2 Czego potrzebujesz 4
3 Wgranie pliku 5
4 Co pojawi się w Home Assistant 6
5 Sprawdzenie, czy działa 7
6 Podstawy bezpieczeństwa 8
7 Kiedy coś nie działa 9
A Dodatek: dashboard i mapa 10
B Dodatek: dane, które wysyła aplikacja 11

NIE MUSISZ ROZUMIEĆ WSZYSTKIEGO

Rozdziały 1–5 wystarczą, żeby to działało. Dodatki na końcu są dla osób, które chcą wystawić kartę na dashboardzie albo napisać własną konfigurację zamiast gotowego pliku.

TO NIE JEST INTEGRACJA

Home Assistant przyjmuje dane z aplikacji przez webhook — funkcja wbudowana, bez instalowania czegokolwiek z zewnątrz. Cała konfiguracja to jeden plik YAML i restart.

ROZDZIAŁ 1

Szybki start — sześć kroków

Całość zajmuje około dziesięciu minut. Potrzebujesz telefonu z aplikacją, dostępu do plików Home Assistant i chwili, żeby uruchomić go ponownie.

1. **Sprawdź adres.** Home Assistant musi być osiągalny przez `https://` także poza domem — np. `https://twoj-ha.pl` albo adres Nabu Casa. Bez tego dane dotrą tylko wtedy, gdy telefon jest w domowej sieci.
2. **Pobierz plik** `driptronic-ha.yaml` ze strony DripTronic (sekcja Materiały).
3. **Wgraj go** do katalogu `packages/` w konfiguracji Home Assistant.
4. **Sprawdź konfigurację i zrestartuj:** Narzędzia deweloperskie → YAML → Sprawdź konfigurację, potem Ustawienia → System → Uruchom ponownie.
5. **Wpisz dane w aplikacji:** Ustawienia → Paliwo i synchronizacja → Home Assistant. Adres HTTPS i Webhook ID `olejarka-trip`, włącz „Wysyłanie danych”.
6. **Przejedź się** kilkaset metrów. Po chwili pojawi się pozycja olejarki i jej prędkość.

MASZ JUŻ SWOJĄ KONFIGURACJĘ OLEJARKI?

Jeśli w `packages/` leży już plik z webhookami olejarki, nie dokładaj drugiego — pojawi się błąd `duplicate unique_id`. Weź z pobranego pliku tylko ten blok, którego ci brakuje (bloki są opisane komentarzami w pliku).

Szczegóły każdego kroku są w rozdziałach 2 i 3. Wszystko inne — dashboard, nazwy pól, test z komputera — znajdziesz w dodatkach na końcu.

ROZDZIAŁ 2

Czego potrzebujesz

Adres, który działa poza domem

Telefon w trasie musi mieć drogę do twojego Home Assistant. Wystarczy jedno z tych rozwiązań:

SPOSÓB	JAK TO WYGLĄDA
Nabu Casa	Adres w stylu `https://xxxx.ui.nabu.casa` — płatna usługa, ale nie wymaga żadnej konfiguracji.
Własna domena	Tunel (np. Cloudflare) albo reverse proxy z certyfikatem: `https://ha.twoja-domena.pl`.
VPN	Telefon z włączonym tunellem (WireGuard, Tailscale) widzi sieć domową jak u siebie.

ADRES LOKALNY = DANE TYLKO W DOMU

Jeśli wpiszesz adres działający wyłącznie w domowej sieci (np. `https://homeassistant.local:8123`), aplikacja wyśle dane tylko wtedy, gdy telefon jest w domu. W trasie wszystko zostanie w kolejce i rozejdzie się po powrocie. To nie jest błąd — to konsekwencja adresu.

Tylko HTTPS

Aplikacja nie wysyła połączeń nieszyfrowanych: adres z `http://` zostanie odrzucony przez Androida, nawet w sieci domowej. Jeśli Home Assistant działa lokalnie po HTTP, wystaw go przez HTTPS.

Trzy rzeczy w aplikacji

W Ustawienia → Paliwo i synchronizacja → Home Assistant potrzebne są tylko: adres, Webhook ID i przełącznik wysyłki. **Interwał** (5–120 s, domyślnie 30 s) decyduje, jak często lecą punkty z jazdy.

Czego nie potrzebujesz

- HACS, dodatku ani custom componentu.
- Konta GitHub ani repozytorium.
- Tokenu ani logowania — webhook je pomija.
- Otwierania portu 8123 na świat, jeśli używasz Nabu Casa lub tunelu.

ROZDZIAŁ 3

Wgranie pliku

Krok po kroku

1. Pobierz `driptronic-ha.yaml` ze strony DripTronic (sekcja Materiały).
2. Otwórz konfigurację Home Assistant — przez File Editor, Studio Code Server albo Samba.
3. Wgraj plik do katalogu `packages/`. Jeśli go nie ma, najpierw dodaj w `configuration.yaml`:

```
homeassistant:
  packages: !include_dir_named packages
```

4. Narzędzia deweloperskie → YAML → **Sprawdź konfigurację**.
5. Ustawienia → System → **Uruchom ponownie**.

Po restarcie encje istnieją, ale są puste (`unknown`). Wypełnią się przy pierwszym odebranym punkcie — czyli wtedy, gdy pojadziesz.

MOŻESZ ZMIENIĆ NAZWY ENCJI

Wszystkie encje nazywają się „Olejarka …”. Możesz zmienić `name:` na własne — zrób to przed pierwszym restartem, wtedy historia zbiera się pod właściwą nazwą. Nie zmieniaj `unique_id:` — to on łączy encję z jej historią.

WEBHOOK ID TO NIE JEST NAZWA DO POKAZANIA ŚWIATU

Plik używa identyfikatorów `olejarka-trip`, `olejarka-trip-end` i `olejarka_refuel`. Każdy, kto zna twój adres i ten identyfikator, może podsłać fałszywe dane — dlatego warto zamienić je na własne, losowe ciągi (w pliku i w aplikacji musi być to samo).

Jeśli masz już część tej konfiguracji

Plik zawiera trzy niezależne bloki, każdy opisany komentarzem: **A** — jazda na żywo, **B** — koniec dnia jazdy, **C** — tankowania. Możesz używać jednego, dwóch albo wszystkich trzech. Usuń z pliku bloki, które już masz gdzie indziej, i zostaw nazwy plików, których nie ruszasz.

ROZDZIAŁ 4

Co pojawi się w Home Assistant

Aplikacja wysyła trzy rodzaje danych, osobno i niezależnie od siebie. Nie musisz ich włączać oddzielnie — wystarczy adres i włączona wysyłka.

KIEDY	CO DOSTAJESZ	SZCZEGÓŁY
podczas jazdy	pozycja + telemetria	Pozycja na mapie, prędkość, obroty, obciążenie, przepustnica, temperatura silnika, poziom oleju w olejarce, chwilowe spalanie, przechył. Jeden punkt na interwał (domyślnie 30 s).
po dniu jazdy	podsumowanie tripu	Dystans, czas, czas w ruchu, średnia i maksymalna prędkość, maksymalny przechył, znacznik końca.
po zapisaniu tankowania	ostatnie tankowanie	Litry, cena za litr, koszt, przebieg, rzeczywiste l/100 km, dystans od poprzedniego tankowania, koszt paliwa z 30 dni.

Łącznie 24 encje: jedna pozycja (`device_tracker.olejarka_gps`) i 23 czujniki. Gotowy plik nazywa je „Olejarka …”, więc łatwo je znaleźć — Narzędzia deweloperskie → Stany, filtr `olejarka`.

> **TO NIE SĄ ENCJE URZĄDZENIA**
> Czujniki z webhooka nie pojawiają się na karcie żadnego urządzenia — są samodzielne, bo nie ma integracji, która mógłby je grupować. Nie jest to błąd i nie da się tego obejść bez własnej integracji.

Kiedy coś zostaje puste

- **Temperatura, obroty, spalanie** — wymagają podłączonego OBD. Bez niego pola po prostu nie istnieją w danych, a encja pokazuje `unknown`.
- **Pozycja** — wymaga zgody na lokalizację w aplikacji.
- **Na postoju** — aplikacja wyłącza GPS, więc punkty lecą tylko w jeździe.
- **Bez zasięgu** — dane czekają w kolejce na telefonie (do 600 punktów) i wychodzą, gdy sieć wróci.

ROZDZIAŁ 5

Sprawdzenie, czy działa

Najszybszy test to zwykła jazda. Nie trzeba nic zmieniać w konfiguracji ani instalować dodatkowych narzędzi.

1. Włącz **Wysyłanie danych** w aplikacji.
2. Przejedź kilkaset metrów i jedź co najmniej tyle, ile wynosi interwał (domyślnie 30 s).
3. W Home Assistant: Narzędzia deweloperskie → **Stany**, filtr `olejarka`.
4. Pozycja i prędkość powinny mieć wartości. Jeśli dalej `unknown` — patrz rozdział 7.

Pod polem adresu w aplikacji widać status ostatniej wysyłki. To pierwsze miejsce, które warto otworzyć, gdy coś nie idzie.

> **TANKOWANIE BEZ JAZDY**
> Możesz sprawdzić kanał tankowań od razu: zapisz dowolne tankowanie w aplikacji (albo edytuj istniejące i zapisaj ponownie). Sensor „ostatnie litry” zmieni wartość bez wyjeżdżania z podjazdu.

Sprawdzenie z komputera (opcjonalnie)

Jedno polecenie wysyła przykładowy punkt do webhooka i pokazuje, czy Home Assistant go przyjmuje:

```
curl -X POST https://twoj-ha.pl/api/webhook/olejarka-trip \
  -H "Content-Type: application/json" \
  -d '{"lat":52.2297,"lon":21.0122,"acc":8,"spd":0}'
```

Poprawna odpowiedź to `200 OK`, a `device_tracker.olejarka_gps` przeskakuje na podane współrzędne. Pomiń ten krok, jeśli nie używasz wiersza poleceń — jazda sprawdzi to samo.

ROZDZIAŁ 6

Podstawy bezpieczeństwa

WEBHOOK ID TO HASŁO

Znajomość adresu `https://twoj-ha.pl/api/webhook/olejarka-trip` pozwala każdemu podsłać fałszywą pozycję. Nie publikuj swojego identyfikatora — zamień go na własny, losowy ciąg (w pliku YAML i w aplikacji, te same).

Warto pamiętać

- **HTTPS** — bez niego dane i tak nie wyjadą z telefonu, a ruch byłby czytelny w sieci.
- **`local_only: false`** — tylko wtedy, gdy naprawdę chcesz przyjmować dane z internetu. Gotowy plik ma to ustawione; zostaw wyłącznie z konieczności.
- **Bez tokenu w adresie** — webhooki nie potrzebują tokenu; nie doklejaj go „na wszelki wypadek”.
- **Zakres danych** — w Home Assistant ląduje pozycja, prędkość, temperatury, poziom oleju i podsumowania. Historia trasy zostaje w aplikacji (eksport GPX), pełny ślad nie jest zapisywany w encjach.

Gdy identyfikator wyciekł

1. Zmień `webhook_id` w pliku i uruchom Home Assistant ponownie — stary adres przestanie przyjmować dane.
2. Wpisz nowy identyfikator w aplikacji.
3. Przejrzyj historię encji; obce dane widać jako skok pozycji albo wartości poza zakresem.

Zmiana `webhook_id` nie rusza encji ani ich historii — encje są łączone przez `unique_id`, którego nie trzeba zmieniać.

Wyłączanie

Wyłączenie wysyłki w aplikacji zatrzymuje ruch natychmiast, bez utraty danych w historii. Usunięcie pliku z `packages/` i restart usuwa encje.

ROZDZIAŁ 7

Kiedy coś nie działa

OBJAW	CO ZWYKLE OZNACZA I JAK TO NAPRAWIĆ
Nic nie przychodzi z trasy, a w domu działa	Adres jest osiągalny tylko w sieci domowej albo `local_only: true`. Użyj adresu z internetu (Nabu Casa, domena, VPN) i ustaw `false`.
Błąd `404` przy teście	Webhook o takim ID nie istnieje: literówka w aplikacji albo brak restartu po wgraniu pliku.
Błąd `duplicate unique_id`	Ten sam blok leży w dwóch plikach w `packages/`. Zostaw jeden.
Encje są puste (`unknown`)	Webhook jeszcze nie przyszedł albo pole nie było w danych — temperatura i obroty wymagają OBD, pozycja zgody na lokalizację.
Brak pozycji na mapie	Sprawdź zgodę na lokalizację w aplikacji; bez GPS punkt nie zawiera współrzędnych.
Sensory zmieniają się tylko w domu	Kolejka na telefonie czeka na sieć. Po powrocie do domu dane wychodzą same.
Błąd szablonu w logach	Nieaktualna składnia albo własna zmiana w pliku. Ustawienia → System → Logi, filtr „webhook”.
Punkty rzadko, choć interwał 5 s	Aplikacja nie wysyła w czasie postoju i oszczędza GPS — to normalne.

Trzy miejsca, które warto otworzyć

- **Logi** — Ustawienia → System → Logi (filtr „webhook”).
- **Stany** — Narzędzia deweloperskie → Stany, filtr `olejarka`.
- **Sprawdzenie konfiguracji** — Narzędzia deweloperskie → YAML, przed każdym restartem.

> **NIE MUSISZ OGARNIAĆ SAM**
> Napisz, co widać w logach i jaki masz adres (bez webhook ID) — postaram się pomóc. Instrukcja aplikacji: `instrukcja.pdf` · plik konfiguracyjny: `driptronic-ha.yaml` · DripTronic 0.6.146 · by rumunn95

DODATEK A

Dashboard i mapa

Karty tworzysz w edytorze dashboardu: Dodaj kartę → Ręczny (YAML). Poniższy układ pokazuje pozycję na mapie i najważniejsze liczby obok.

```
type: vertical-stack
cards:
  - type: map
    entities:
      - entity: device_tracker.olejarka_gps
    hours_to_show: 2
    default_zoom: 14
  - type: glance
    columns: 3
    entities:
      - entity: sensor.olejarka_predkosc
        name: Prędkość
      - entity: sensor.olejarka_obroty
        name: Obroty
      - entity: sensor.olejarka_temp_silnika
        name: Silnik
```

Nazwy encji wynikają z nazw w pliku — Home Assistant zamienia polskie znaki na ASCII, więc „Olejarka prędkość” to `sensor.olejarka_predkosc`. Jeśli zmienisz nazwę w YAML-u, sprawdź identyfikator w Narzędziach deweloperskich → Stany.

Historia i wykresy

Czujniki mają klasę pomiaru, więc działają w kartach historii i w statystykach długoterminowych; możesz je dodać też do panelu Energia. Karta mapy korzysta z kafelków OpenStreetMap i nie wymaga klucza API. Ślad trasy z `device_tracker` jest ograniczony Retencją historii (domyślnie kilka dni).

PEŁNE TRASY

Pełny ślad trasy nie jest zapisywany w encjach — celowo, bo rozdmuchałby bazę. Trasę masz w aplikacji i w eksporcie GPX.

DODATEK B

Dane, które wysyła aplikacja

Przydatne, jeśli chcesz samodzielnie pisać szablony albo dodać własne encje. W polach JSON dostajesz tylko to, co jest dostępne — brak klucza nie jest błędem.

Jazda na żywo — `olejarka-trip`

POLE	JEDNOSTKA	ZNACZENIE
lat, lon	°	Pozycja GPS (6 miejsc po przecinku).
acc	m	Dokładność pozycji.
spd	km/h	Prędkość.
hdg	°	Kierunek ruchu.
fuel	L	Paliwo w baku (jeśli OBD podaje).
tmp	°C	Temperatura silnika.
tilt	°	Maksymalny przechył z dnia.
rpm	obr/min	Obroty silnika.
load, throttle	%	Obciążenie silnika i położenie przepustnicy.
oil	ml	Poziom oleju w olejarce.
fuelh	L/h	Chwilowe zużycie paliwa.

Koniec dnia — `olejarka-trip-end`

POLE	JEDNOSTKA	ZNACZENIE
day	—	Dzień tripu.
start / end	ISO 8601	Początek i koniec jazdy.
dist_km	km	Dystans.
dur_min	min	Czas od startu do końca, z postojami.
moving_min	min	Czas faktycznej jazdy.
avg_kmh	km/h	Średnia liczona z czasu w ruchu.
max_kmh	km/h	Prędkość maksymalna.
max_tilt	°	Maksymalny przechył.
status	—	`open` w trakcie, `closed` po zakończeniu.

Tankowanie — `olejarka_refuel`

POLE	JEDNOSTKA	ZNACZENIE

data	ISO 8601	Data i godzina wpisu.
liters	L	Ile zatankowano.
price	zł/L	Cena za litr (jeśli podana).
cost	zł	Koszt tankowania (jeśli podany).
odo	km	Przebieg z OBD.
l100	l/100 km	Rzeczywiste spalanie z OBD.
since	km	Dystans od poprzedniego tankowania.
cost30	zł	Koszt paliwa z ostatnich 30 dni.
partial	—	`true` dla dotankowania (nie do pełna).

Dokument opisuje konfigurację po stronie Home Assistant dla aplikacji DripTronic 0.6.146. W razie rozbieżności wiążące są nazwy pól w logach webhooka i aktualna dokumentacja Home Assistant.