Resmon – monitor

Podręcznik użytkownika

Spis treści

[1. Opis 1](#_Toc517326638)

[1.1. Wymagania 1](#_Toc517326639)

[1.1.1. Wymagania sprzętowe 1](#_Toc517326640)

[1.1.2. Wymagania systemowe 1](#_Toc517326641)

[2. Instalacja 2](#_Toc517326642)

[2.1. Opcje instalatora 2](#_Toc517326643)

[3. Instrukcja użytkowania 3](#_Toc517326644)

[3.1. Uruchomienie 3](#_Toc517326645)

[3.2. Plik konfiguracyjny połączenia z bazą danych 3](#_Toc517326646)

[4. Informacje dla programistów 3](#_Toc517326647)

[4.1. Używane moduły Pythona 3](#_Toc517326648)

[4.2. Skrypty dla programistów 4](#_Toc517326649)

[4.3. Wydawanie aplikacji w kontenerze Docker 4](#_Toc517326650)

[5. Autorzy 5](#_Toc517326651)

# Opis

Program ResMon-monitor jest częścią systemu ResMon. Służy odbierania pomiarów z sensorów oraz udostępniania ich klientom (GUI oraz automatycznemu).

**Bardzo ważne!** Podane plik klucza (*./data/jwt.key*) oraz konfiguracja połączenia z bazą danych (*./data/db.conf.json*) są używane tylko w środowisku testowym. Proszę absolutnie nie używać ich w wersji produkcyjnej.

**Uwaga!** Wszystkie ścieżki wymienione w tym dokumencie są traktowane, jakby były używane w głównym katalogu projektu/instalacji.

## Wymagania

### Wymagania sprzętowe

Program nie posiada specjalnych wymagań sprzętowych.

### Wymagania systemowe

Program został stworzony wyłącznie pod system operacyjny Linux z rodziny Ubuntu, Debian, Mint, itd.

Do poprawnego działania instalatora oraz samej aplikacji konieczne jest posiadanie zainstalowanego:

1. interpretera języka **Python** w wersji **co najmniej 3.6**[[1]](#footnote-1);
2. odpowiedniej dla niej wersji managera pakietów **PIP**[[2]](#footnote-2);
3. dostępnej z miejsca uruchomienia monitora instancji bazy danych MongoDB[[3]](#footnote-3).

# Instalacja

Program dla zwykłych użytkowników jest dystrybuowany jako pojedynczy plik instalatora  
*install-monitor.sh*. Jest to skrypt powłoki Bash. W celu instalacji aplikacji ResMon-monitor należy:

1. pobrać najnowszą wersję instalatora z repozytorium: <https://github.com/MAD-FiS/resmon-monitor/releases> ;
2. przenieść go do lokalizacji, w której ma zostać zainstalowany klient automatyczny;
3. uruchomić instalator komendą:  
    ./install-monitor.sh --config CONFIG\_FILE --key KEY\_FILE [--quiet]
4. udzielić zgody instalatorowi na rozpakowanie plików;
5. poczekać do końca instalacji.

Opcje instalatora zostały opisane w sekcji Opcje instalatora.

Plik konfiguracyjny połączenia z bazą danych oraz klucz JWT zostaną skopiowane do wnętrza monitora, dlatego nawet ich późniejsze usunięcie nie spowoduje problemów z działaniem programu.

**Uwaga!** Ewentualne ostrzeżenia lub błędy wyświetlone podczas instalacji mogą spowodować nieuruchomienie się aplikacji lub jej niepoprawne działanie.

## Opcje instalatora

Instalator posiada kilka parametrów, które zostały przedstawione w tabeli 1.

| **Opcja** | **Wartość domyślna** | **Opis** |
| --- | --- | --- |
| --config CONFIG\_FILE | --- | lokalizacja pliku z konfiguracją połączenia z bazą danych, który zostanie skopiowany do monitora |
| --key KEY\_FILE | --- | lokalizacja pliku z kluczem JWT, który zostanie skopiowany do monitora |
| --quiet | false | włącza tzw. tryb cichy instalacji; w tym trybie automatycznie następuje akceptacja wszystkich zgód, o które normalnie instalator pyta użytkownika i czeka na jego odpowiedź; może być zastosowane w przypadku, gdy nie jest możliwa interakcja programu z użytkownikiem podczas procesu instalacji |

Tabela Opcje instalatora

# Instrukcja użytkowania

## Uruchomienie

Aby rozpocząć pracę z programem, należy go uruchomić, wywołując następującą komendę w miejscu, w którym został on zainstalowany:

./resmon-monitor [--stop]

W tabeli 2 zostały opisane możliwe opcje, jakie może przyjmować program.

| **Opcja** | **Wartość domyślna** | **Opis** |
| --- | --- | --- |
| --stop | false | po ustawieniu tej opcji monitor zostanie zatrzymany, zamiast być uruchomiony |

Tabela Opis możliwych parametrów uruchomieniowych programu

REST-owe API będzie dostępne na porcie 81, a moduł odbierający dane od sensora – na porcie 82.

Przerwanie wykonywania programu następuje poprzez uruchomienie skryptu *./resmon-monitor* z flagą *--stop*.

**Uwaga!** Plik z kluczem musi być taki sam jak na monitorach, które używają tego serwera uwierzytelniającego do potwierdzenia tożsamości użytkowników. Po wygenerowaniu klucza należy sprawdzić czy zarówno serwer autoryzacyjny, ja i monitory używają tej samej wersji tego pliku.

## Plik konfiguracyjny połączenia z bazą danych

Plik konfiguracyjny połączenia z bazą danych to plik w formacie JSON. Jego pola zostały opisane w tabeli 3.

| **Nazwa** | **Typ** | **Opis** |
| --- | --- | --- |
| address | string | Adres, pod którym jest dostępna baza danych MongoDB |
| port | string | Port, na którym nasłuchuje MongoDB |
| user | string | Użytkownik używany do połączenia się z bazą danych |
| password | string | Hasło używane do połączenia się z bazą danych |

Tabela Pola pliku konfiguracyjnego połączenia z bazą danych

# Informacje dla programistów

**Uwaga!** Ta instrukcja jest napisana tylko dla programistów, którzy używają systemu operacyjnego Linux.

Projekt jest utrzymywany na serwerze Github pod następującym adresem: <https://github.com/MAD-FiS/resmon-monitor> . Przed rozpoczęciem pracy należy sklonować powyższe repozytorium.

W przypadku pracy nad repozytorium nie jest wymagana instalacja, opisana w sekcji Instalacja.

## Używane moduły Pythona

Te moduły są wymagane przez aplikację. Jeśli potrzebne jest na przykład uruchomienie testów, należy się upewnić, że wszystkie one zostały zainstalowane na komputerze przy pomocy programu *PIP3*. Można użyć do tego pliku *./data/requirements* .

Lista modułów znajduje się w tabeli 4.

| **Nazwa modułu** | **Wersja** |
| --- | --- |
| connexion | 1.1.15 |
| coverage | 4.0.3 |
| flask\_jwt\_extended | 3.7.2 |
| flask\_testing | 0.6.1 |
| nose | 1.3.7 |
| pluggy | 0.3.1 |
| py | 1.4.31 |
| pymongo | 3.6.1 |
| python\_dateutil | 2.6.0 |
| randomize | 0.13 |
| setuptools | 21.0.0 |

Tabela 4 Moduły Pythona używane przez program

## Skrypty dla programistów

Dla programistów, którzy chcieliby rozwijać aplikację, przygotowano kilka skryptów, pozwalających usprawnić pracę. Uruchamia się je, wywołując w głównym katalogu programu komendę:

./scripts.sh SCRIPT\_NAME

gdzie SCRIPT\_NAME może być jedną z następujących opcji:

* **build** – przygotowuje plik instalatora *install-monitor.sh*;
* **docgen** – generuje dokumentację kodu i umieszcza ją w katalogu *./docs/* ;
* **runtest** – uruchamia wszystkie testy, dostępne dla tego projektu.

**Uwaga!** Wszystkie elementy aplikacji automatycznie wczytują potrzebne środowisko. Jeśli trzeba zrobić to ręcznie, plik z tą konfiguracją znajduje się w katalogu ./data/ .

## Wydawanie aplikacji w kontenerze Docker

Aplikacja może być uruchamiana w ramach kontenera Docker’owego. Może to być wykorzystane do testowania jej w czystym, izolowanym środowisku.

Na komputerze musi być zainstalowane oprogramowanie Docker[[4]](#footnote-4). Przed rozpoczęciem tworzenia obrazu kontenera należy się upewnić, iż w katalogu głównym projektu znajduje się  
plik *install-monitor.sh*. Jeśli nie, należy wcześniej uruchomić skrypt, budujący instalator, opisany w sekcji Instalacja.

Do poprawnego uruchomienia systemu przydatne będzie wcześniejsze stworzenie kontenera z bazą danych MongoDb. Robi się to następującą komendą:

docker run --name resmon-monitor-db -d mongo

Następnie trzeba zbudować obraz kontenera komendą:

docker build -t resmon-monitor .

oraz uruchomić go:

docker run -p 4000:81 -p 4001:82 --link resmon-monitor-db -it resmon-monitor

Kontener zostanie uruchomiony razem z aplikacją na nim działającą jako usługa w ramach serwera Apache. REST-owe API jest dostępne na porcie 4000, a moduł odbierający dane od sensora – na porcie 4001.

# Autorzy

Autorami projektu ResMon jest grupa MAD-Team[[5]](#footnote-5) w składzie:

1. Bajorek Tomasz – kierownik projektu, programista;
2. Bartocha Łukasz – specjalista z zakresu Dev Ops;
3. Chronowski Tomasz – programista;
4. Drzazga Kamil – programista;
5. Kwaśnicki Marcin – programista;
6. Rucki Grzegorz – programista;
7. Ryniak Grzegorz – kierownik techniczny, programista;
8. Szęszoł Rafał – programista;
9. Słoczyński Tomasz – programista.

1. <https://docs.python.org/3.6/> [↑](#footnote-ref-1)
2. <https://pypi.org/project/pip/> [↑](#footnote-ref-2)
3. <https://www.mongodb.com/> [↑](#footnote-ref-3)
4. <https://docs.docker.com/> [↑](#footnote-ref-4)
5. <https://github.com/MAD-FiS> [↑](#footnote-ref-5)