

# Dokumentacja projektu strony Flask

Uniwersytet WSB Merito Oddział w Chorzowie

Autorzy:

Tomasz Duka

Mateusz Krajewski

Daniel Kasperek

## 1) Opis projektu

Napisana przez nas aplikacja webowa w Pythonie z wykorzystaniem frameworka Flask umożliwia:

- rejestrację i logowanie użytkowników
- uwierzytelnianie sesji
- wizualizację danych z pliku CSV w postaci wykresów
- wybór kategorii oraz lokalizacji danych do analizy

## 2) Wykorzystane technologie

Lista wykorzystanych technologii:

- Python 3
- Flask– framework webowy
- Flask-Login– obsługa logowania i sesji
- Flask-SQLAlchemy– ORM do bazy danych
- Flask-Bcrypt– haszowanie haseł
- SQLite
- Pandas– przetwarzanie danych CSV
- Matplotlib– generowanie wykresów
- HTML / CSS– interfejs użytkownika

### 3) Struktura projektu

#### PythonFlask

- 1) app.py
- 2) config.py
- 3) extensions.py
- 4) plotter.py
- 5) requirements.txt
- 6) README.md
- 7) data
  - data.csv
- 8) atabase
  - models.py
  - user\_queries.py
- 9) static
  - css
    - auth.css
    - style.css
  - diagrams
    - plot.png
- 10) templates
  - login.html
  - register.html
  - dashboard.html

## 4) Konfiguracja aplikacji

Plik config.py zawiera konfigurację aplikacji:

- SECRET\_KEY – klucz bezpieczeństwa Flask
- SQLALCHEMY\_DATABASE\_URI – ścieżka do bazy danych
- SQLALCHEMY\_TRACK\_MODIFICATIONS – wyłączone dla wydajności

## 5) Rozszerzenia Flask

Plik extensions.py inicjalizuje:

- db – SQLAlchemy
- login\_manager – Flask-Login
- bcrypt – haszowanie haseł

## 6) Baza danych

Modele

Plik database/models.py definiuje model 'User', który przechowuje:

- id
- username
- password

## Zapytania użytkownika

Plik database/user\_queries.py zawiera funkcje:

- create\_user() – tworzenie nowego użytkownika
- get\_user\_by\_username() – pobieranie użytkownika
- check\_user\_password() – weryfikacja hasła

## 7) Obsługa użytkowników

Aplikacja umożliwia:

- Rejestrację– formularz register.html
- Logowanie– formularz login.html
- Wylogowanie– endpoint /logout

Hasła są bezpiecznie haszowane przy użyciu bcrypt.

## 8) Wizualizacja danych

Plik plotter.py odpowiada za:

- wczytywanie danych z data/data.csv
- pobieranie dostępnych kategorii i lokalizacji,
- generowanie wykresów zapisywanych w static/diagrams/plot.png

Wykresy są następnie wyświetlane na stronie dashboard.html

## 9) Routing aplikacji

Najważniejsze endpointy w app.py:

- /login – logowanie użytkownika
- /register – rejestracja użytkownika
- /dashboard – panel główny z wykresami (wymaga logowania)
- /logout – wylogowanie

## 10) Instalacja i uruchomienie

1. Rozpakuj projekt

2. Utwórz wirtualne środowisko:

- |                             |               |
|-----------------------------|---------------|
| 1) python -m venv venv      |               |
| 2) source venv/bin/activate | Dla Linux/Mac |
| 3) venv\Scripts\activate    | Dla Windows   |

3. Zainstaluj zależności:

- pip install -r requirements.txt

4. Uruchom aplikację:

- python app.py

5. Otwórz przeglądarkę:

- <http://localhost:8000>