

## Projekt 2 (PJN) - Generator zdań

Prosta aplikacja webowa oparta o Flask, która generuje angielskie zdania w różnych czasach i trybach na podstawie wybranych elementów (podmiot, czasownik, dopełnienie).

The screenshot shows a web application titled "English SVO Builder" with the instruction "Wybierz parametry, aby zbudować zdanie." (Choose parameters to build a sentence). The interface is divided into three main sections: 1. Subject (S), 2. Verb (V), and 3. Object (O). Each section contains several dropdown menus for selection. In the Subject section, the choices are Determiner: "my", Adjective: "beautiful", Noun: "dog", and Number: "Plural (>1)". In the Verb section, the choices are Verb: "eat", Tense: "Future Simple", and Mode: "Twierdzenie (Affirmative)". In the Object section, the choices are Determiner: "(none)", Adjective: "green", Noun: "apple", and Number: "Plural (>1)". Below these sections is a dark blue button labeled "Generuj zdanie". At the bottom, a light blue box displays the result: "Wynik: My beautiful dogs will eat green apples."

Figure 1: Podgląd aplikacji

### O aplikacji

Aplikacja pozwala składać poprawne zdania po angielsku, łącząc podmiot, czasownik i dopełnienie, a następnie odmieniając czasownik i dobierając rodzajniki. Wspiera podstawowe czasy i tryby, więc nadaje się do nauki gramatyki oraz szybkiego testowania przykładów.

### Wymagania

- Python 3.10+
- (opcjonalnie) GNU Make

## Szybki start

1. Utwórz środowisko i zainstaluj zależności:

```
make install
```

2. Uruchom aplikację:

```
make run
```

Serwer wystartuje lokalnie pod adresem `http://127.0.0.1:5000`.

## Uruchomienie bez Makefile

```
python -m venv .venv
.\.venv\Scripts\pip install -U pip
.\.venv\Scripts\pip install -e .
.\.venv\Scripts\python app.py
```

## Tryb developerski (Flask)

```
make dev
```

## API

POST `/generate`

Przykładowe wywołanie:

```
curl -X POST http://127.0.0.1:5000/generate ^
-H "Content-Type: application/json" ^
-d '{"subject_noun":"cat","subject_number":"singular",\
  "verb":"eat","object_noun":"apple",\
  "object_number":"singular","tense":"present",\
  "mode":"affirmative"}'
```

Odpowiedź:

```
{"sentence":"The cat eats an apple."}
```

## Struktura projektu

- `app.py` - główna aplikacja Flask i routing
- `engine.py` - logika składania zdania i odmiany czasowników
- `data.py` - dane wejściowe (czasowniki, rzeczowniki, przymiotniki)
- `templates/` - szablony HTML
- `static/` - CSS i JS

## Dostępne cele Makefile

- `make install` - tworzy venv i instaluje zależności
- `make run` - uruchamia aplikację
- `make dev` - uruchamia serwer developerski Flask
- `make clean` - usuwa środowisko wirtualne