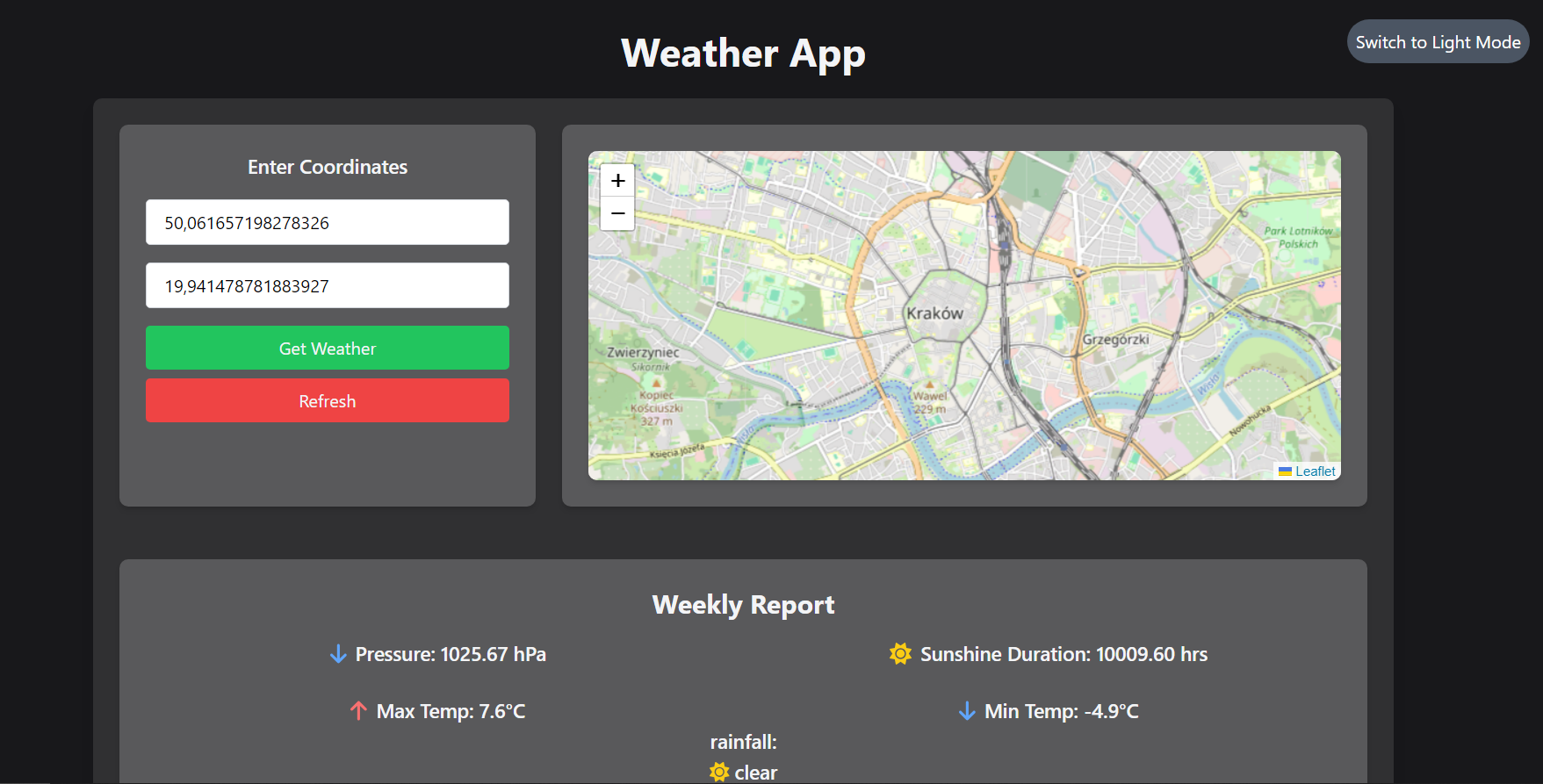
**Dokumentacja Komponentu Home - Aplikacja Pogodowa**

**Opis Aplikacji**

Aplikacja pogodowa umożliwia użytkownikowi uzyskanie prognozy pogody na podstawie współrzędnych geograficznych (szerokości i długości geograficznej). Po podaniu współrzędnych użytkownik może uzyskać szczegółowe prognozy na podstawie dwóch źródeł API:

* **Dane godzinowe**: dostarczają informacje o temperaturze dla każdej godziny.
* **Dane tygodniowe**: zapewniają podsumowanie pogody na przestrzeni siedmiu dni.

Dodatkowo aplikacja obsługuje **tryb jasny** i **tryb ciemny**, co pozwala na dostosowanie interfejsu do preferencji użytkownika.



**Endpointy**

**1. Endpoint: fetchWeather**

**Cel funkcji**

Pobranie danych pogodowych dla wybranej lokalizacji na podstawie współrzędnych geograficznych. Funkcja oblicza:

* Maksymalne i minimalne temperatury dla każdego dnia.
* Opisy warunków pogodowych na podstawie kodów pogodowych.
* Szacowaną produkcję energii słonecznej (w kWh) na podstawie długości nasłonecznienia.

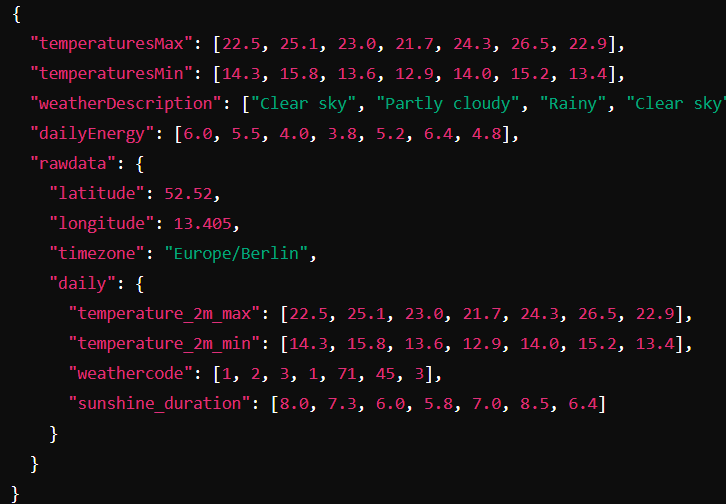
**Parametry wejściowe**

* **latitude**: szerokość geograficzna (liczba).
* **longitude**: długość geograficzna (liczba).

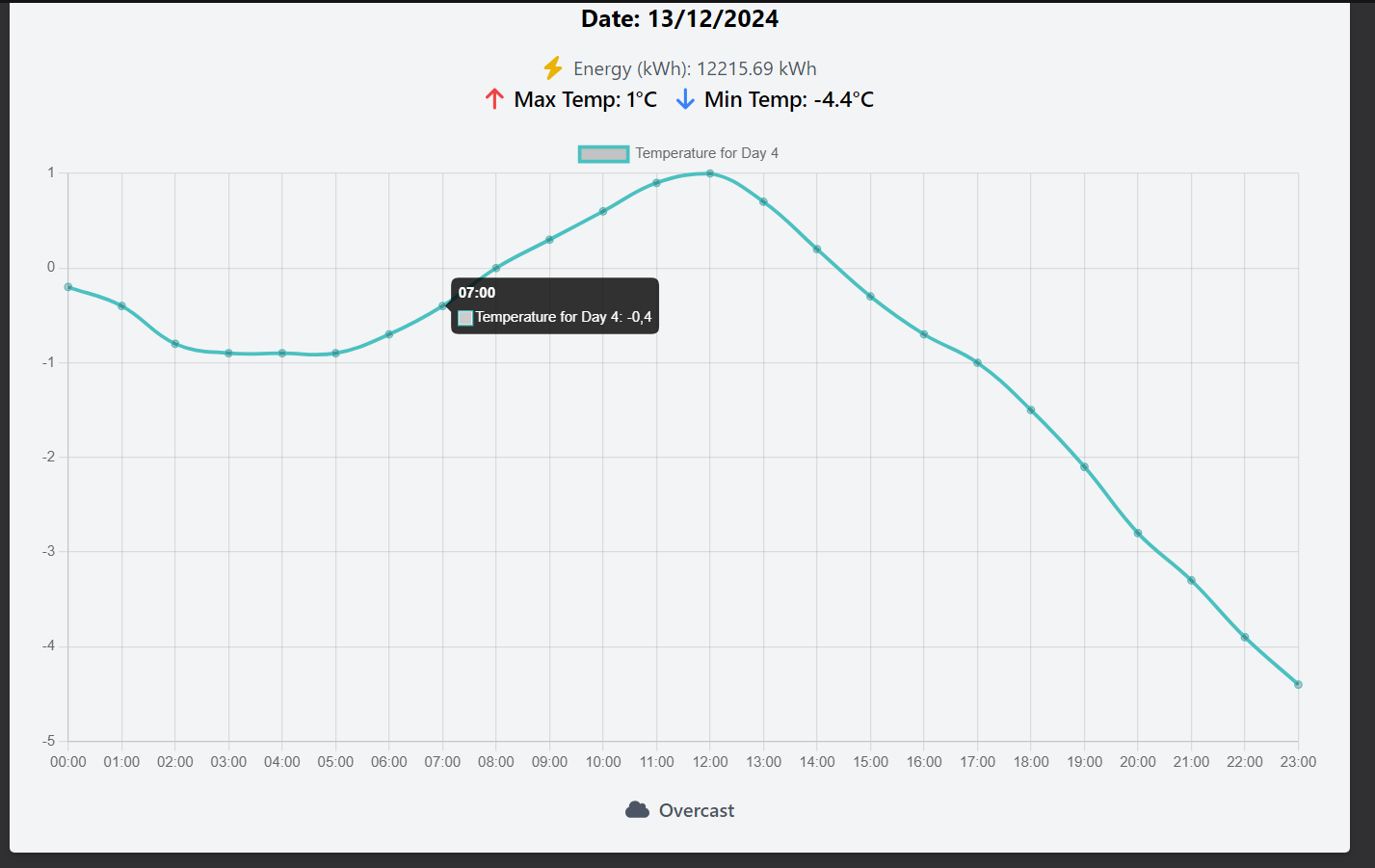
**Proces działania**

1. **Walidacja danych wejściowych**: Sprawdzenie poprawności współrzędnych.
2. **Pobranie danych z API Open-Meteo**:
   * Budowanie dynamicznego adresu URL na podstawie podanych współrzędnych.
   * Pobranie danych o temperaturze i nasłonecznieniu.
3. **Przetwarzanie danych**:
   * Obliczenie maksymalnej i minimalnej temperatury dla każdego dnia.
   * Tłumaczenie kodów pogodowych na opisy warunków atmosferycznych.
   * Oszacowanie energii słonecznej na podstawie wzoru: Energia(kWh)=Moc panelu×Sprawnosˊcˊ panelu×Długosˊcˊ nasłonecznieniaEnergia (kWh) = Moc\ panelu \times Sprawność\ panelu \times Długość\ nasłonecznieniaEnergia(kWh)=Moc panelu×Sprawnosˊcˊ panelu×Długosˊcˊ nasłonecznienia
   * Przyjęte wartości: moc panelu = **2.5 kW**, sprawność = **20%**.

**Zwracane dane**

****

* **temperaturesMax**: Tablica maksymalnych temperatur dla każdego dnia.
* **temperaturesMin**: Tablica minimalnych temperatur dla każdego dnia.
* **weatherDescription**: Opisowe warunki pogodowe dla każdego dnia.
* **dailyEnergy**: Szacowana produkcja energii (w kWh) dla każdego dnia.
* **rawdata**: Surowe dane z API.



**2. Endpoint: fetchWeeklySummary**

**Cel funkcji**

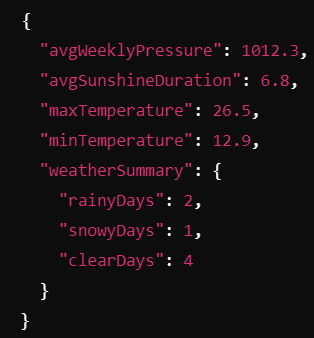
Pobranie prognozy tygodniowej i przetworzenie jej na podsumowanie:

* Średnie ciśnienie atmosferyczne.
* Średni czas nasłonecznienia.
* Maksymalne i minimalne temperatury.
* Podsumowanie pogody na podstawie liczby dni deszczowych i śnieżnych.

**Proces działania**

1. **Walidacja danych wejściowych**: Sprawdzenie poprawności współrzędnych.
2. **Pobranie danych z API Open-Meteo**.
3. **Obliczenia**:
   * Średnie wartości ciśnienia i nasłonecznienia.
   * Maksymalne i minimalne temperatury tygodniowe.
   * Podsumowanie pogodowe: 'rainy', 'snowy', 'clear' na podstawie liczby dni deszczowych i śnieżnych.

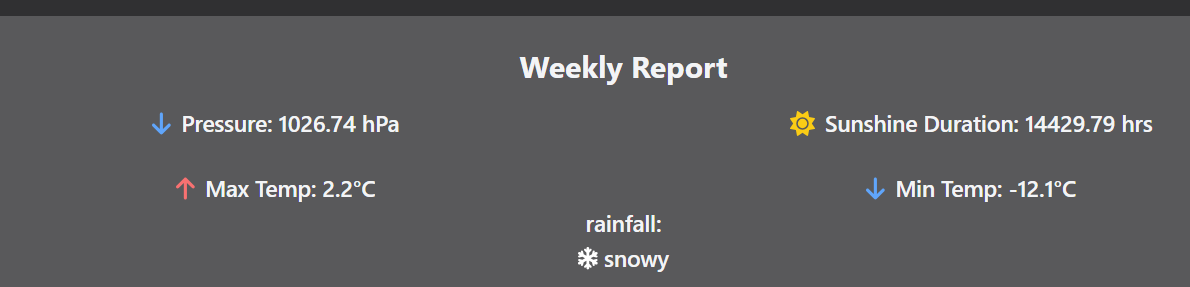
**Zwracane dane**

****

* **avgWeeklyPressure**: Średnie ciśnienie atmosferyczne (hPa).
* **avgSunshineDuration**: Średni czas nasłonecznienia (godziny).
* **maxTemperature**: Maksymalna temperatura (°C).
* **minTemperature**: Minimalna temperatura (°C).
* **weatherSummary**: Podsumowanie pogody ('rainy', 'snowy', 'clear').

3 Różne typy pogody oceniane przez aplikacje:







**Struktura Komponentu**

**1. Sekcje interfejsu**

1. **Formularz wejściowy**: Umożliwia wprowadzenie współrzędnych geograficznych.
2. **Mapa**: Dynamiczna mapa lokalizacji użytkownika.
3. **Raport tygodniowy**: Podsumowanie pogody w skali tygodnia.
4. **Dane godzinowe**: Wykresy temperatur, nasłonecznienia oraz warunków pogodowych.

**2. Kluczowe stany komponentu**

* **latitude i longitude**: Współrzędne geograficzne wprowadzone przez użytkownika.
* **weatherDataHourly**: Dane godzinowe (np. temperatury).
* **maxTemp i minTemp**: Maksymalne i minimalne temperatury dla każdego dnia.
* **avgWeeklyPressure i avgWeekSunshineDuration**: Średnie wartości ciśnienia i nasłonecznienia w tygodniu.
* **isLightMode**: Flaga trybu wyświetlania (jasny/ciemny).

**Główne Funkcje**

**1. handleFetchWeather**

Wywoływana po kliknięciu przycisku "Get Weather". Obsługuje pobieranie danych z API, aktualizację stanów komponentu oraz zarządzanie błędami.

**2. refreshPage**

Resetuje stany aplikacji, pozwalając na ponowne wprowadzenie współrzędnych.

**3. getChartData**

Generuje dane dla wykresów temperatur.

**4. getWeatherIcon**

Zwraca odpowiednią ikonę pogodową na podstawie warunków atmosferycznych.

**5. toggleMode**

Przełącza tryb wyświetlania (jasny/ciemny).

**Zrealizowane cele:**

**odsumowanie realizacji zadania:**

**1. Backend**

**Zrealizowano:**

* **Pierwszy endpoint (fetchWeather):**
  + **Pobiera prognozę pogody na 7 dni z Open-Meteo API.**
  + **Zwraca dane:**
    - **Data.**
    - **Kod pogody.**
    - **Minimalna i maksymalna temperatura.**
    - **Szacowana produkcja energii w kWh, obliczona według wzoru**
* **Drugi endpoint (fetchWeeklySummary):**
  + **Oblicza i zwraca:**
    - **Średnie ciśnienie w ciągu tygodnia.**
    - **Średni czas ekspozycji na słońce.**
    - **Minimalną i maksymalną temperaturę.**
    - **Podsumowanie tygodnia (z opadami/bez opadów).**

**Dodatkowe aspekty:**

* **Walidacja danych wejściowych (szerokość i długość geograficzna, zakres wartości).**
* **Obsługa błędów zewnętrznego API (np. brak połączenia, nieprawidłowe dane).**
* **Wyniki zwracane w formacie JSON.**
* **Bardziej szczegółowa walidacja danych wejściowych, np. sprawdzanie czy współrzędne są liczbami w prawidłowym zakresie.**

**Do dalszej realizacji:**

* **Testy automatyczne (jeśli czas pozwoli, zostaną dodane).**

**2. Frontend**

**Zrealizowano:**

* **Tabela wyświetlająca prognozę pogody na 7 dni:**
  + **Data w formacie DD/MM/YYYY.**
  + **Ikona pogody (zintegrowano z FontAwesome na podstawie kodów pogody z API).**
  + **Minimalna i maksymalna temperatura.**
  + **Szacowana produkcja energii w kWh dla każdego dnia.**
* **Stopka z ogólnymi informacjami:**
  + **Skrajne temperatury tygodnia.**
  + **Średnie ciśnienie w ciągu tygodnia.**
  + **Średni czas ekspozycji na słońce.**
  + **Komentarz podsumowujący prognozę (z opadami/bez opadów).**

**Dodatkowe aspekty:**

* **Responsywność.**
* **Automatyczne pobieranie lokalizacji użytkownika za pomocą Geolocation API.**
* **Dark mode.**
* **Integracja z mapą świata (Leaflet) dla wyboru lokalizacji.**

**Podsumowanie**

Komponent Home stanowi centralny element aplikacji pogodowej, umożliwiając użytkownikom interaktywne uzyskiwanie i wizualizowanie danych pogodowych. Aplikacja integruje szczegółowe dane pogodowe z funkcjami mapowania i wizualizacji, tworząc nowoczesny i użyteczny interfejs użytkownika.