

# Wybrana technologia, narzędzia wspierające produkcję oprogramowania, product backlog, funkcjonalności sprint 1.

## 1) Wybrana technologia

### Założenia ogólne:

Oprogramowanie rozwijane będzie jako narzędzie/aplikacja internetowa w celu zapewnienia dostępu jak największej grupie odbiorców (użytkowników końcowych) niezależnie od sprzętu i systemu operacyjnego, z którego korzystają.

### Front-end:

Front-end aplikacji zostanie wykonany przy użyciu otwartego frameworku Angular (ostatnia stabilna wersja 13.1.1) oraz języka programowania JavaScript. Do tworzenia wykresów użyta zostanie biblioteka ngx-charts współpracująca z Angularem. Wybrano ją ze względu na łatwość użycia, jakość powstających wykresów oraz fakt, że ich dodatkowe cechy w tym interaktywność podlegają customizacji.

### Back-end:

Back-end aplikacji zostanie stworzony w Pythonie z wykorzystaniem zapytań restowych i biblioteki json do obróbki pliku konfiguracyjnego i pobranych ze strony datahub'a danych (w postaci pliku json).

## 2) Narzędzia wspierające produkcję oprogramowania.

### System kontroli wersji:

Jako system kontroli wersji zostanie wykorzystany github ze względu na prostotę użycia i dobrą znajomość przez zespół.

### Kontrola i zarządzanie pracą:

Do kontroli i rozdziału wykonywanych zadań wykorzystano darmową wersję aplikacji jira (link: <https://www.atlassian.com/software/jira>).

### 3) Product backlog

#### Sprint 1:

- Must
  - Podgląd danych technicznych (Koszt: 8),
  - Odświeżanie danych (Koszt: 5)
- Should
  - Definiowanie typów wykresu (Koszt: 8),
  - Prostota pliku konfiguracyjnego (Koszt: 2)
- Could
- Would
  - Wiele funkcji na jednym wykresie (Koszt 5),
  - Automatyczne odświeżanie (Koszt 5)

#### Sprint 2:

- Must
  - Wybór różnych plików konfiguracyjnych (Koszt: 21),
  - Czytelne wyświetlanie wielu wykresów (Koszt: 34),
  - Ograniczona pamięć (Koszt: 5),
  - Definiowanie jednostek i formatów (Koszt 8),
  - Skalowalność aplikacji (Koszt: 21)
- Should
  - Przełączanie pomiędzy wykresami (Koszt: 21),
  - Definiowanie zakresu wartości (Koszt: 5),
  - Częstotliwość odświeżania (Koszt 5)
- Could
- Would
  - Zamiana jednostek (Koszt: 13),
  - Dostosowanie wyglądu do użytkownika (Koszt: 13),
  - Zależność wyglądu od konfiguracji (Koszt: 13)

#### Sprint 3:

- Must
  - Szybkość działania (Koszt: ?)
- Should
  - Ograniczenie okresu (Koszt: 5)
- Could
  - Szybkość uruchamiania (Koszt: ?),
  - Ograniczenie liczby zgłaszanych błędów (Koszt 2)
- Would
  - Zmiana okresu (Koszt 21)

#### **4) Funkcjonalności na sprint 1**

W ramach pierwszego sprintu planowane jest zapewnienie następujących funkcjonalności tworzonej aplikacji:

- Pobranie informacji z datahub'a,
- Opracowanie formatu pliku konfiguracyjnego,
- Stworzenie uproszczonej wersji parsera do pliku konfiguracyjnego,
- Zapewnienie komunikacji pomiędzy frontendem i backendem aplikacji (plik .csv),
- Stworzenie uproszczonej wersji frontendu aplikacji (możliwość wyświetlania pojedynczego wykresu na podstawie pobranych danych i informacji z pliku konfiguracyjnego oraz wyświetlanie aktualnych wartości wybranych parametrów).