

Politechnika Warszawska

WYDZIAŁ ELEKTRONIKI  
I TECHNIK INFORMACYJNYCH



Instytut Automatyki i Informatyki Stosowanej

# Praca dyplomowa inżynierska

na kierunku Informatyka  
w specjalności Sztuczna Inteligencja

Aplikacja do analizy danych GNSS zebranych od piłkarzy

Aleksandra Szymańska

Numer albumu 318733

promotor  
dr hab. inż. Mariusz Kamola

WARSZAWA 2026



## Aplikacja do analizy danych GNSS zebranych od piłkarzy

**Streszczenie.** W pracy zaprojektowano i zaimplementowano aplikację internetową do analizy danych z globalnych systemów nawigacji satelitarnej (GNSS, ang. *Global Navigation Satellite System*) zbieranych od piłkarzy. Rozwiązanie odpowiada na ograniczoną możliwość konfiguracji parametrów analizy w systemach komercyjnych. Aplikacja umożliwia wczytywanie plików CSV eksportowanych z platformy STATSports, filtrowanie i agregację danych wejściowych, wykrywanie odcinków biegu z wysoką prędkością (HSR, ang. *High-Speed Running*) oraz zdarzeń przyspieszenia i hamowania, wyznaczanie profili przyspieszenie–prędkość (ASP, ang. *Acceleration–Speed Profile*) i hamowanie–prędkość (DSP, ang. *Deceleration–Speed Profile*), analizę aktywności w strefach boiska, interaktywną prezentację oraz eksport wyników. Użytkownik może zmieniać parametry filtrowania, wykrywania zdarzeń oraz wyznaczania profili. Pozwala to na ocenę intensywności aktywności zawodników i porównywanie wyników uzyskiwanych przy różnych ustawieniach analizy. Elementem wyróżniającym system jest profil DSP, wykorzystujący podejście analogiczne do profilu ASP w analizie hamowań. Uwzględnienie parametrów mechanicznych oraz podziału zdarzeń na fazę wczesną i późną umożliwia weryfikację teorii dotyczącej przebiegu hamowania w praktyce. System działa w architekturze klient–serwer i wykorzystuje React oraz FastAPI. Poprawność implementacji zweryfikowano testami jednostkowymi i integracyjnymi warstwy serwerowej, testami warstwy klienckiej, porównaniem z referencyjną implementacją profilu ASP oraz ręczną weryfikacją end-to-end na rzeczywistych danych.

**Słowa kluczowe:** aplikacja internetowa, analiza danych sportowych, GNSS, hamowanie, piłka nożna, profil przyspieszenie–prędkość

## **Application for analysis of GNSS data collected from football players**

**Abstract.** This thesis presents the design and implementation of a web application for analysing Global Navigation Satellite System (GNSS) data collected from football players. The application addresses the limited configurability of analysis parameters in commercial systems. It enables importing CSV files exported from the STATSports platform, filtering and aggregation of input data, detection of high-speed running (HSR) sections, acceleration and deceleration events, calculation of acceleration–speed profile (ASP) and deceleration–speed profile (DSP), pitch zone analysis, interactive visualisation and export of results. Users can modify filtering, event detection and profile estimation parameters. This supports assessment of player activity intensity and comparison of results obtained with different analysis settings. A key contribution of the system is the DSP profile, which applies the ASP profiling approach to deceleration analysis. Mechanical parameters and the division of events into early and late phases make it possible to evaluate the theory describing the course of deceleration in practice. The system uses a client–server architecture with React and FastAPI. The application was verified using server-side unit and integration tests, client-side tests, comparison with a reference implementation of the ASP method and manual end-to-end verification on real data.

**Keywords:** acceleration–speed profile, deceleration, football, GNSS, sports data analysis, web application

# Spis treści

|                                                          |    |
|----------------------------------------------------------|----|
| <b>1. Wstęp</b>                                          | 7  |
| 1.1. Motywacja                                           | 7  |
| 1.2. Cel                                                 | 8  |
| <b>2. Przegląd literatury i istniejących rozwiązań</b>   | 9  |
| 2.1. Analiza danych GNSS w literaturze                   | 9  |
| 2.1.1. Przetwarzanie sygnału                             | 9  |
| 2.1.2. Bieg z wysoką prędkością                          | 10 |
| 2.1.3. Profil przyspieszenie–prędkość                    | 11 |
| 2.1.4. Przyspieszenia i hamowania                        | 12 |
| 2.1.5. Analiza mechaniczna hamowań                       | 12 |
| 2.2. Komercyjne systemy GNSS/IMU                         | 12 |
| 2.2.1. STATSports                                        | 13 |
| 2.2.2. Catapult                                          | 13 |
| 2.2.3. gpexe                                             | 14 |
| 2.2.4. Podsumowanie i ograniczenia systemów komercyjnych | 14 |
| <b>3. Wymagania</b>                                      | 15 |
| 3.1. Wymagania niefunkcjonalne                           | 15 |
| 3.2. Wymagania funkcjonalne                              | 15 |
| 3.2.1. Uwierzytelnianie                                  | 15 |
| 3.2.2. Konfiguracja parametrów                           | 15 |
| 3.2.3. Wczytywanie i przygotowanie danych                | 15 |
| 3.2.4. Analiza przestrzenna i HSR                        | 16 |
| 3.2.5. Analiza przyspieszeń i hamowań                    | 16 |
| 3.2.6. Prezentacja wyników                               | 16 |
| 3.2.7. Eksport wyników                                   | 17 |
| <b>4. Projekt i implementacja systemu</b>                | 18 |
| 4.1. Architektura systemu                                | 18 |
| 4.1.1. Komponenty warstwy klienckiej                     | 19 |
| 4.1.2. Komponenty warstwy serwerowej                     | 25 |
| 4.2. Model danych                                        | 26 |
| 4.2.1. Dane wejściowe                                    | 26 |
| 4.2.2. Kontrakt wymiany danych klient–serwer             | 27 |
| 4.2.3. Wyjście systemu                                   | 29 |
| 4.3. Zastosowane technologie                             | 30 |
| 4.3.1. React i biblioteki warstwy klienckiej             | 30 |
| 4.3.2. Python i biblioteki analityczne                   | 31 |
| 4.3.3. FastAPI i Pydantic                                | 31 |
| 4.3.4. Amazon Web Services                               | 31 |
| 4.4. Implementacja algorytmów analitycznych              | 32 |
| 4.4.1. Filtr Butterwortha                                | 32 |

|                                                                       |           |
|-----------------------------------------------------------------------|-----------|
| 4.4.2. Profile przyspieszenie–prędkość i hamowanie–prędkość . . . . . | 32        |
| 4.4.3. Analiza stref boiska . . . . .                                 | 34        |
| 4.4.4. Wykrywanie okien czasowych . . . . .                           | 36        |
| 4.5. Bezpieczeństwo systemu . . . . .                                 | 37        |
| <b>5. Weryfikacja rozwiązania . . . . .</b>                           | <b>39</b> |
| 5.1. Testy jednostkowe i integracyjne warstwy serwerowej . . . . .    | 39        |
| 5.2. Porównanie z referencyjną implementacją profilu ASP . . . . .    | 39        |
| 5.3. Testy warstwy klienckiej . . . . .                               | 40        |
| 5.4. Weryfikacja end-to-end . . . . .                                 | 40        |
| <b>6. Podsumowanie . . . . .</b>                                      | <b>41</b> |
| <b>Bibliografia . . . . .</b>                                         | <b>43</b> |
| <b>Wykaz symboli i skrótów . . . . .</b>                              | <b>45</b> |
| <b>Spis rysunków . . . . .</b>                                        | <b>46</b> |
| <b>Spis tabel . . . . .</b>                                           | <b>46</b> |

# 1. Wstęp

Globalne systemy nawigacji satelitarnej (GNSS, ang. *Global Navigation Satellite System*) są powszechnie wykorzystywane w sportach zespołowych [5], [19]. Zbierane przez nie informacje o chwilowym położeniu i prędkości umożliwiają wyznaczenie wskaźników opisujących aktywność, obciążenie i formę sportowców. Niewielkie rozmiary odbiorników GNSS sprawiają, że mogą być noszone w formie kamizelek, opasek lub innych akcesoriów bez istotnego ograniczania swobody ruchu.

Współcześnie trener nie musi opierać decyzji dotyczących przygotowania i powoływania zawodników wyłącznie na subiektywnej obserwacji. Dane zbierane podczas treningu i rywalizacji sportowej pozwalają na ilościową i bardziej obiektywną ocenę aktywności zawodników. W odróżnieniu od sztucznie przygotowanych testów wysiłkowych odzwierciedlają one rzeczywiste zachowania zawodników w warunkach meczowych i treningowych [12], [19]. Wyznaczane na ich podstawie wskaźniki umożliwiają między innymi ocenę obciążenia, monitorowanie postępów i zmęczenia, indywidualizację obciążeń treningowych, analizę zachowań taktycznych oraz wspieranie procesu powrotu do zdrowia po urazach [5], [19].

## 1.1. Motywacja

W trakcie pojedynczej sesji aktywności gromadzone są duże ilości danych. Dla jednego zawodnika liczba odczytów z sensorów może sięgać kilkuset tysięcy w ciągu godziny. Wyzwaniem staje się ich sprawne przetwarzanie i interpretacja.

Producenci systemów monitorowania oferują własne platformy do analizy rejestrowanych danych. Gotowe rozwiązania udostępniają narzucony zestaw metod analizy, co ogranicza ich wykorzystanie w sytuacjach wymagających zastosowania niestandardowych technik.

Część procedur stosowanych w systemach komercyjnych jest niejawna. Dotyczy to między innymi filtrowania sygnału i sposobu wyznaczania złożonych wskaźników [11], [20]. Użytkownik może konfigurować część parametrów analizy, na przykład strefy prędkości, progi przyspieszenia i hamowania do wykrywania zdarzeń. Nie może zmienić progu minimalnego trwania zdarzenia oraz kryteriów doboru punktów do profili opisujących maksymalne możliwości mechaniczne zawodnika. Z perspektywy badawczej utrudnia to ocenę wpływu przyjętych założeń na uzyskane wyniki.

Eksport danych z platform umożliwia prowadzenie analiz przy użyciu własnych skryptów. Zapewnia to większą elastyczność i wgląd w sposób przetwarzania, ale wymaga wiedzy programistycznej i jest mniej wygodne niż korzystanie z dedykowanej aplikacji umożliwiającej import plików, konfigurację parametrów oraz interaktywną eksplorację wyników.

Przedstawione ograniczenia wskazują na potrzebę opracowania rozszerzalnej platformy analitycznej, która połączy przejrzystość obliczeń z możliwością konfiguracji specjalistycznych metod analizy.

### 1.2. Cel

Celem pracy było zaprojektowanie i implementacja aplikacji umożliwiającej analizę danych GNSS zbieranych od piłkarzy. System wspiera ocenę intensywności aktywności zawodników i porównywanie wyników uzyskiwanych dla różnych parametrów analizy.

Aplikacja została opracowana na potrzeby klubu sportowego z myślą o jej dalszym rozwoju. Obsługuje dane pochodzące z urządzeń firmy STATSports. Innymi założeniami projektu są dostępność z poziomu przeglądarki internetowej, ograniczenie ryzyka związanego z przetwarzaniem danych zawodników i niski koszt utrzymania systemu.

System umożliwia filtrowanie i agregację danych wejściowych, wyznaczanie profilu przyspieszenie–prędkość (ASP, ang. *Acceleration–Speed Profile*) [12] oraz interaktywną prezentację wyników analizy.

Elementem wyróżniającym projekt jest niestandardowy zestaw wskaźników i konfigurowalnych metod, obejmujący między innymi wyznaczanie profilu hamowanie–prędkość (DSP, ang. *Deceleration–Speed Profile*). Publicznie dostępny kod źródłowy pozwala prześledzić sposób działania zaimplementowanych algorytmów.

## 2. Przegląd literatury i istniejących rozwiązań

### 2.1. Analiza danych GNSS w literaturze

GNSS obejmuje między innymi amerykański GPS, europejski Galileo, rosyjski GLONASS i chiński BeiDou. Odbiorniki wyznaczają pozycję na podstawie sygnałów docierających z wielu satelitów. Wykorzystanie kilku konstelacji zwiększa liczbę dostępnych satelitów i może poprawiać jakość oraz ciągłość wyznaczania pozycji [13]. Dokładność pomiaru może być ograniczana przez przeszkody terenowe, zabudowę i konstrukcje stadionowe [14].

Wartości wskaźników obliczanych na podstawie danych GNSS mogą różnić się w zależności od wcześniejszych etapów pozyskiwania i przetwarzania danych, w tym oceny jakości sygnału, filtrowania danych, sposobu obliczania przyspieszenia oraz klasyfikacji aktywności do stref intensywności. Zależność ta jest szczególnie istotna przy wyznaczaniu profili ASP i analizie krótkotrwałych fragmentów aktywności. Sposób filtrowania może wpływać na dobór progów oraz pozostałych parametrów analizy [11], [20].

#### 2.1.1. Przetwarzanie sygnału

Na etapie pomiaru istotne znaczenie mają częstotliwość próbkowania odbiornika oraz jakość sygnału satelitarnego. Zaleca się stosowanie urządzeń rejestrujących dane z częstotliwością co najmniej 10 Hz [14]. O jakości pomiarów GNSS informują między innymi liczba wykorzystywanych satelitów i współczynnik HDOP (ang. *Horizontal Dilution of Precision*), opisujący wpływ ich rozmieszczenia na dokładność lokalizacji. Wartość HDOP zbliżona do 1 oznacza korzystny układ [11].

Dane GNSS zawierają szumy pomiarowe wynikające między innymi z wielodrogowości sygnału i błędów wyznaczania pozycji. W celu ich redukcji stosuje się na przykład średnią ruchomą, filtr medianowy i filtr Butterwortha czwartego rzędu o częstotliwości odcięcia 2 Hz [11], [20].

Średnia ruchoma wygładza sygnał poprzez zastąpienie każdej próbki średnią arytmetyczną wartości z zadanego okna. Metoda ta jest jednak wrażliwa na wartości skrajne, ponieważ każda próbka z okna jest uwzględniana w obliczeniach. Filtr medianowy działa na podobnej zasadzie, wykorzystując medianę wartości z okna, dzięki czemu ogranicza wpływ pojedynczych obserwacji odstających.

Filtr Butterwortha jest filtrem dolnoprzepustowym typu IIR (ang. *Infinite Impulse Response*), co oznacza, że wartość sygnału wyjściowego zależy od bieżących i wcześniejszych próbek sygnału wejściowego oraz wcześniejszych wartości sygnału wyjściowego. Cechą filtrów IIR jest wprowadzanie przesunięcia fazowego zależnego od częstotliwości, co może prowadzić do przesunięcia w czasie zdarzeń analizowanych na sygnale wyjściowym.

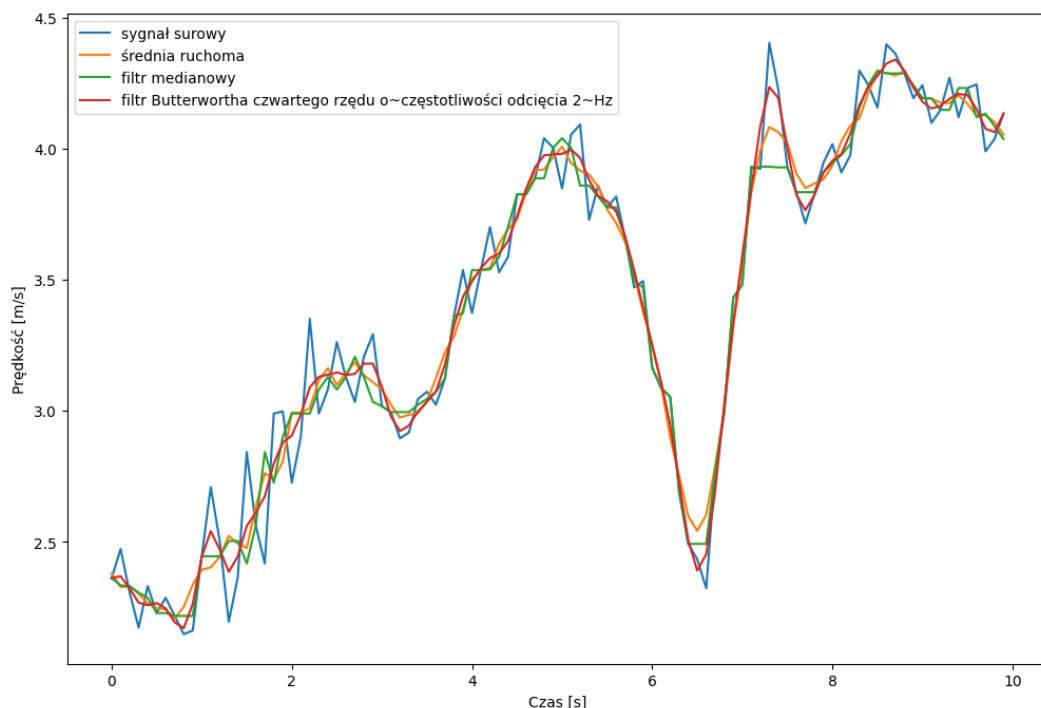
Filtr Butterwortha  $N$ -tego rzędu ma charakterystykę amplitudową postaci:

$$|H(j\omega/\omega_{3dB})| = \frac{1}{\sqrt{1 + (\omega/\omega_{3dB})^{2N}}},$$

gdzie  $N$  oznacza rząd filtru, a  $\omega_{3dB}$  pulsację, przy której moc sygnału spada do połowy wartości z pasma przepustowego, co odpowiada tłumieniu o około 3 dB.

Płaska charakterystyka w paśmie przepustowym pozwala zachować przebieg zmian prędkości, a tłumienie składowych powyżej częstotliwości odcięcia ogranicza wpływ szumu pomiarowego. Wyższy rząd filtru oznacza silniejsze tłumienie składowych powyżej  $\omega_{3dB}$  [21].

Porównanie działania omówionych filtrów na fragmencie sygnału prędkości pochodzącego z pomiaru GNSS przedstawiono na rysunku 2.1.



**Rysunek 2.1.** Porównanie średniej ruchomej, filtru medianowego i filtru Butterwortha na sygnale prędkości GNSS.

Filtr medianowy najsilniej wygładza sygnał, ale w przebiegu widoczne są krótkie fragmenty o stałej wartości. Średnia ruchoma silnie tłumi szybkie zmiany, ale wpływ wartości skrajnych przenosi się na sąsiednie próbki. Filtr Butterwortha zachowuje kształt rzeczywistych zmian prędkości i łagodnie tłumi szum wysokoczęstotliwościowy.

Filtrowanie jest szczególnie istotne podczas wyznaczania przyspieszenia na podstawie zmian prędkości w czasie, ponieważ różniczkowanie uwypatnia błędy pomiarowe.

### 2.1.2. Bieg z wysoką prędkością

Do opisu intensywnych fragmentów aktywności zawodnika wykorzystuje się między innymi bieg z wysoką prędkością (HSR, ang. *High-Speed Running*). Zdarzenia tego typu definiuje się zwykle jako fragmenty, w których prędkość zawodnika przekracza ustalony próg przez co najmniej określony czas. Warunek minimalnego czasu trwania ogranicza wpływ pojedynczych, krótkotrwałych przekroczeń progu.

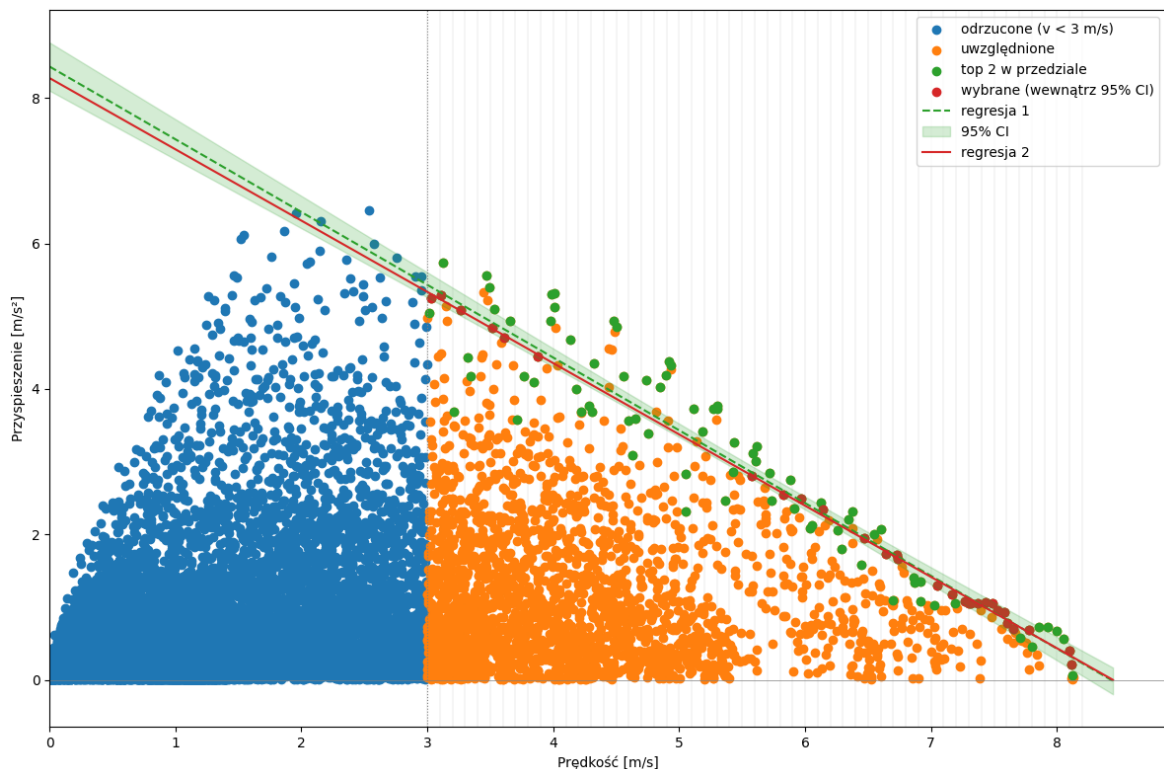
W piłce nożnej mężczyzn jako próg HSR często przyjmuje się wartości zbliżone do 19–20 km/h. Badano również minimalny czas trwania zdarzenia, testując wartości od 0,1

do 1 s. W literaturze nie istnieją jednak powszechnie przyjęte wartości tych parametrów, a stosowane definicje różnią się pomiędzy badaniami, dyscyplinami sportowymi oraz producentami systemów monitorowania zawodników [6], [20].

### 2.1.3. Profil przyspieszenie–prędkość

Profil ASP opisuje maksymalne możliwości przyspieszania zawodnika przy różnych prędkościach biegu. Punktem wyjścia jest obserwacja, że najwyższe wartości przyspieszenia osiągane są przy niskich prędkościach, natomiast w pobliżu prędkości maksymalnej możliwości dalszego przyspieszania stają się coraz mniejsze. Zależność tę można opisać modelem liniowym [12].

Profil ASP wraz z kolejnymi etapami jego wyznaczania przedstawiono na rysunku 2.2.



**Rysunek 2.2.** Profil ASP wraz z etapami jego wyznaczania.

Wyznaczanie profilu rozpoczyna się od odrzucenia próbek o prędkości poniżej 3 m/s. Pozostały zakres prędkości dzielony jest na przedziały o szerokości 0,2 m/s, a z każdego z nich wybierane są dwie próbki o najwyższych wartościach przyspieszenia. Do wybranych punktów dopasowywana jest regresja liniowa. Obserwacje wykraczające poza 95% przedział ufności regresji są następnie odrzucane, a na pozostałych punktach wyznaczana jest końcowa regresja opisująca profil ASP. Wyraz wolny tej regresji odpowiada teoretycznemu maksymalnemu przyspieszeniu, natomiast jej miejsce zerowe odpowiada teoretycznej prędkości maksymalnej [12].

### 2.1.4. Przyspieszenia i hamowania

W sportach zespołowych przyspieszenia i hamowania odgrywają istotną rolę ze względu na częste zmiany prędkości zawodników podczas gry. Wykrywa się je jako zdarzenia, w których wartość przyspieszenia lub opóźnienia przekracza ustaloną granicę przez co najmniej określony czas. Kryterium zakończenia zdarzenia może być na przykład spadek wartości przyspieszenia do  $0 \text{ m/s}^2$ , poniżej wartości rozpoczynającej zdarzenie albo określonej jej części. Stosowane progi wykrywania mają zwykle wartość bezwzględną od około 2,5 do  $3,0 \text{ m/s}^2$ , a minimalny czas trwania wynosi od 0,2 do 1 s [7], [20].

Zdarzenia klasyfikuje się do stref intensywności na podstawie osiąganych wartości przyspieszenia. Wartości bezwzględne przekraczające  $2,5 \text{ m/s}^2$  i  $3,5 \text{ m/s}^2$  stosowane są odpowiednio jako progi wysokiej i bardzo wysokiej intensywności [7]. Stosowanie niewielkiej liczby szerokich stref może jednak prowadzić do utraty części informacji o intensywności wykonywanych działań.

### 2.1.5. Analiza mechaniczna hamowań

Gwałtowne hamowania były dotychczas rzadziej opisywane, mimo że takie działania występują częściej niż intensywne przyspieszenia i nakładają na zawodnika większe obciążenie mechaniczne. Hamowanie zwykle poprzedza zmianę kierunku ruchu lub zatrzymanie, natomiast przyspieszanie częściej odbywa się w linii prostej i wymaga dłuższego narastania prędkości. Podczas hamowania zawodnik musi w krótkim czasie rozproszyć energię kinetyczną zgromadzoną podczas biegu [7], [8].

Analiza hamowań oparta wyłącznie na wartości ujemnego przyspieszenia nie wystarcza do oceny obciążenia mechanicznego [8]. Na podstawie danych GNSS oraz masy ciała zawodnika można wyznaczać siłę, moc i impuls.

Wskaźniki mechaniczne hamowania można wyznaczać dla całego zdarzenia lub osobno dla fazy wczesnej i późnej. Granicę pomiędzy fazami wyznacza moment osiągnięcia 50% prędkości z początku hamowania. Faza wczesna ma odpowiadać szybkiej i intensywnej redukcji prędkości, natomiast faza późna mniejszej redukcji prędkości i stabilizacji ruchu. Taki podział pozwala ocenić przebieg hamowania względem przyjętego modelu [8].

## 2.2. Komercyjne systemy GNSS/IMU

Popularność systemów GNSS wynika między innymi z relatywnie niskich kosztów wdrożenia i mobilności urządzeń, pozwalającej na prowadzenie pomiarów podczas codziennych treningów i meczów wyjazdowych.

W badaniach dotyczących monitorowania zawodników w sportach zespołowych wykorzystywane są między innymi komercyjne platformy firm STATSports, Catapult i gpexe [3], [10]. STATSports Apex rejestruje dane z częstotliwością 18 Hz [15], Catapult Vector 8 z częstotliwością 10 Hz [1], natomiast gpexe UNO z częstotliwością 25 Hz [4]. Te rozwiązania są zintegrowane z czujnikami inercyjnymi (IMU, ang. *Inertial Measurement Unit*), w skład których wchodzi akcelerometr, żyroskop i magnetometr. Systemy pozwalają między innymi na analizę dystansu, prędkości, HSR, przyspieszeń i hamowań oraz działań

w zdefiniowanych strefach intensywności. Umożliwiają konfigurowanie raportów, pracę z wybranymi fragmentami aktywności i wieloma sesjami oraz śledzenie trendów.

Poszczególne platformy różnią się między innymi możliwościami definiowania progów i stref intensywności, zastosowaniem danych z IMU, wyznaczanymi wskaźnikami obciążenia i metodami analizy mechanicznej, wizualizacjami aktywności na boisku oraz wykorzystaniem algorytmów AI/ML [1], [4], [18].

### 2.2.1. STATSports

Platforma Sonra umożliwia definiowanie indywidualnych i zespołowych stref intensywności dla prędkości, przyspieszeń, hamowań oraz mocy metabolicznej, szacowanej jako iloczyn chwilowej prędkości i kosztu energetycznego pokonania jednostki dystansu. Producent udostępnia domyślne wartości progów, które mogą być modyfikowane przez użytkownika. Przykładowo domyślny próg HSR wynosi 5,5 m/s.

Dane z IMU wykorzystywane są między innymi do wykrywania asymetrii ruchu i wyznaczania *Total Loading*, czyli syntetycznej miary obciążenia, obliczanej jako suma modułów zmian wektora przyspieszenia podzielona przez czynnik skalujący równy 1000 [18].

Platforma wyznacza profil siła–prędkość (FVP, ang. *Force–Velocity Profile*), służący do oceny maksymalnych możliwości zawodnika do generowania siły poziomej przy różnych prędkościach biegu. Ze względu na brak możliwości bezpośredniego mierzenia siły jest ona wyznaczana na podstawie masy zawodnika i jego przyspieszenia [17]. Przy założeniu implementacji zgodnej z oryginalną metodą ta nie umożliwia konfiguracji progu minimalnej prędkości, szerokości przedziałów, liczby próbek wybieranych w przedziale oraz kryterium odrzucania punktów przed drugą regresją.

System umożliwia wizualizację danych dotyczących położenia zawodników w postaci map ciepła, map pozycji i map rozmieszczenia sprintów na boisku oraz odtwarzanie przebiegu sesji.

Producent deklaruje wykorzystanie AI do analizy trendów oraz w formie asystenta odpowiadającego na pytania dotyczące wyników zawodników, ich zmian w czasie i porównań [16].

### 2.2.2. Catapult

Platforma OpenField umożliwia definiowanie własnych progów i parametrów analizy oraz porównywanie wyników zawodników do wartości referencyjnych wyznaczanych na podstawie danych zespołu. W dokumentacji systemu zdarzenia przyspieszenia i hamowania są domyślnie wykrywane odpowiednio dla wartości powyżej  $2 \text{ m/s}^2$  i poniżej  $-2 \text{ m/s}^2$ , natomiast standardowy próg HSR to 5,5 m/s.

Dane z IMU wykorzystywane są między innymi do wykrywania zmian kierunku, skoków, asymetrii ruchu i kontaktów fizycznych oraz do wyznaczania *Player Load*, czyli syntetycznej miary obciążenia obliczanej jako suma modułów zmian wektora przyspieszenia podzielona przez 100.

System umożliwia wizualizację danych dotyczących położenia zawodników w postaci

map ciepła i śladów ruchu zawodnika oraz analizę pojedynczych sprintów, przyspieszeń i hamowań wraz z ich rozmieszczeniem na boisku.

Producent deklaruje wykorzystanie algorytmów AI i ML, ale nie wskazuje ich konkretnego zastosowania [2].

### 2.2.3. gpexe

Platforma gpexe definiuje progi intensywnych przyspieszeń i hamowań na podstawie indywidualnych maksymalnych możliwości zawodnika.

System wykrywa zdarzenia o wysokim zapotrzebowaniu energetycznym na podstawie przekroczenia progu mocy metabolicznej.

Platforma wyznacza profil ASP, służący do oceny maksymalnych możliwości zawodnika do przyspieszania przy różnych prędkościach biegu. Przy założeniu implementacji zgodnej z oryginalną metodą ta nie umożliwia konfiguracji progu minimalnej prędkości, szerokości przedziałów, liczby próbek wybieranych w przedziale oraz kryterium odrzucania punktów przed drugą regresją. Oblicza wskaźnik *High External Work*, czyli dodatnią pracę mechaniczną we fragmentach aktywności przekraczających próg wysokiej intensywności.

Dane z IMU są wykorzystywane przez algorytmy AI i ML do wykrywania wybranych zdarzeń ruchowych, takich jak skoki, kontakty i uderzenia [4].

### 2.2.4. Podsumowanie i ograniczenia systemów komercyjnych

Rozwiązania komercyjne są rozbudowanymi platformami wspierającymi monitorowanie procesu treningowego. Wyznaczają wiele wskaźników, udostępniają wyniki w formie interaktywnych wizualizacji i mogą umożliwiać konfigurację progów dla HSR, przyspieszeń i hamowań. Zakres analizy oraz możliwość modyfikacji ustawień metod zależą od konkretnego systemu. Deklarowane funkcje AI/ML nie służą bezpośrednio opisowi obciążenia, lecz wspierają interpretację danych.

W platformach stosowane są syntetyczne wskaźniki obciążenia, takie jak Player Load i Total Loading. Pełnią one podobną funkcję, ale różnią się nazwą i sposobem skalowania wartości. Mogą być użyteczne do porównywania obciążeń w ramach jednego systemu, ale nie stanowią bezpośrednich miar fizycznych.

W części platform są dostępne złożone metody, takie jak ASP lub FVP, ale możliwości ich konfiguracji są ograniczone. Dotyczy to między innymi minimalnego czasu trwania zdarzeń, progu minimalnej prędkości, szerokości przedziałów prędkości, liczby próbek wybieranych z przedziału oraz kryterium odrzucania punktów odstających. Uniemożliwia to porównywanie wyników uzyskanych dla różnych ustawień analizy.

Ograniczona jest również przejrzystość stosowanych metod. Kod źródłowy nie jest publiczny, a dokumentacja dla użytkownika nie zawiera szczegółów filtrowania sygnału i doboru próbek, co utrudnia ocenę wpływu przyjętych założeń na uzyskane wyniki.

### **3. Wymagania**

Przedstawione ograniczenia systemów komercyjnych wskazują na potrzebę opracowania rozwiązania umożliwiającego rozbudowaną konfigurację parametrów analizy oraz wykorzystanie specjalistycznych metod analizy hamowań. Zestaw funkcjonalności systemu określono na podstawie narzędzi dostępnych w platformach komercyjnych i rozwiązań opisywanych w literaturze. Wymagania dotyczące obsługi aplikacji i bezpieczeństwa danych określono z uwzględnieniem oczekiwań klubu sportowego.

#### **3.1. Wymagania niefunkcjonalne**

- System powinien być dostępny z poziomu przeglądarki internetowej.
- System nie powinien trwale przechowywać danych wejściowych użytkowników ani wyników analizy.
- System powinien być dostępny tylko dla uwierzytelnionych użytkowników.
- System powinien zapewniać szyfrowanie komunikacji.
- System powinien przetwarzać dane obejmujące setki tysięcy próbek w czasie akceptowalnym dla pracy interaktywnej.
- System powinien mieć publicznie dostępny kod źródłowy.
- System powinien być rozszerzalny o nowe formaty danych, metody analizy i wskaźniki.
- System powinien być utrzymywany w ramach bezpłatnych limitów usług chmurowych.

#### **3.2. Wymagania funkcjonalne**

##### **3.2.1. Uwierzytelnianie**

- System uwierzytelnia użytkowników na podstawie współdzielonego hasła.
- System utrzymuje sesję użytkownika przez ograniczony czas.
- System przekierowuje niezalogowanych użytkowników do formularza logowania.
- System ogranicza liczbę prób logowania z jednego adresu IP.

##### **3.2.2. Konfiguracja parametrów**

- System umożliwia konfigurację parametrów jakości sygnału GNSS, domyślnej masy ciała zawodnika i kolorów prezentacji wyników.
- System umożliwia konfigurację parametrów filtrowania prędkości, wyznaczania profili ASP i DSP, wykrywania HSR oraz zdarzeń przyspieszenia i hamowania.
- System umożliwia wybór jednostek podawanych parametrów prędkości.
- System umożliwia nadpisanie wybranych parametrów dla poszczególnych plików.
- System przechowuje konfigurację użytkownika w pamięci lokalnej przeglądarki.

##### **3.2.3. Wczytywanie i przygotowanie danych**

- System umożliwia wczytanie wielu plików.

- System obsługuje pliki zgodne ze strukturą eksportu STATSports.
- System przetwarza pozostałe pliki mimo błędu w jednym z nich.
- System odrzuca dane niespełniające kryteriów jakości.
- System uśrednia dane o powtórzonych znacznikach czasowych.
- System filtruje pomiary prędkości.
- System wyznacza przyspieszenie na podstawie prędkości.

#### 3.2.4. Analiza przestrzenna i HSR

- System przekształca współrzędne geograficzne na współrzędne boiska i dzieli je na strefy.
- System wykrywa odcinki HSR dla całego boiska i jego stref.
- System wyznacza statystyki aktywności z możliwością ograniczenia do stref boiska i odcinków HSR.
- System wyznacza rozkłady prędkości z możliwością ograniczenia do stref boiska i odcinków HSR.

#### 3.2.5. Analiza przyspieszeń i hamowań

- System wyznacza profile ASP i DSP.
- System wykrywa zdarzenia przyspieszenia i hamowania.
- System wyznacza parametry kinematyczne i mechaniczne zdarzeń oraz ich faz wcześniejszej i późniejszej.
- System zlicza zdarzenia w wybranych przedziałach intensywności, z możliwością ograniczenia do stref boiska i odcinków HSR.
- System agreguje statystyki zdarzeń w wybranych przedziałach intensywności, z możliwością ograniczenia do stref boiska i odcinków HSR.

#### 3.2.6. Prezentacja wyników

- System prezentuje wyniki w tabelach i na interaktywnych wykresach.
- System umożliwia ukrywanie serii wyników.
- System umożliwia wybór między widokiem przyspieszenia i siły.
- System prezentuje prędkość w tabelach w m/s i km/h.
- System umożliwia wybór jednostki prędkości na wykresie szeregu czasowego.
- System umożliwia wybór między czasem względnym i bezwzględnym na wykresach szeregów czasowych.
- System umożliwia wybór między skalą liniową i logarytmiczną na histogramach.
- System umożliwia zaznaczanie próbek na wykresach profili.
- System umożliwia przeglądanie, sortowanie i usuwanie zaznaczonych próbek w dedykowanej tabeli.
- System umożliwia wybór liczby punktów wyróżnianych wokół zaznaczonej próbki na wykresach profili, szeregów czasowych i trajektorii.
- System umożliwia wybór zakresu prezentowanych statystyk aktywności.
- System umożliwia wybór zakresu prezentowanych statystyk zdarzeń.

- System udostępnia stronę z informacjami o systemie, formacie plików i wykorzystanych źródłach.

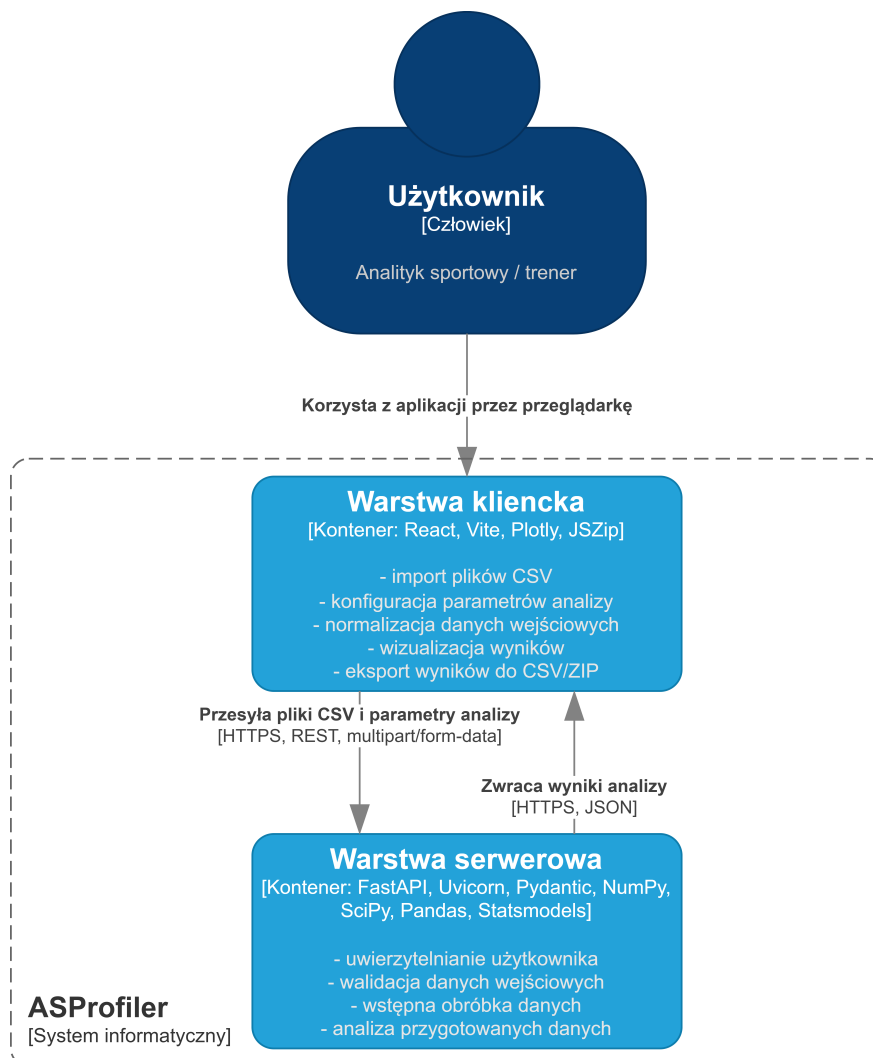
#### **3.2.7. Eksport wyników**

- System umożliwia eksport wyników poszczególnych analiz.
- System umożliwia eksport wszystkich wyników i konfiguracji analizy.

## 4. Projekt i implementacja systemu

### 4.1. Architektura systemu

Ogólną architekturę systemu przedstawiono na rysunku 4.1.



Rysunek 4.1. Diagram C4 kontenerów projektowanego systemu.

System zaimplementowano w architekturze klient–serwer. Warstwę kliencką stanowi aplikacja przeglądarkowa opracowana z wykorzystaniem biblioteki React, natomiast warstwą serwową jest aplikacja FastAPI udostępniająca interfejs REST API. Rozdzielenie warstw umożliwia realizację obliczeń analitycznych po stronie serwera z wykorzystaniem naukowych bibliotek języka Python.

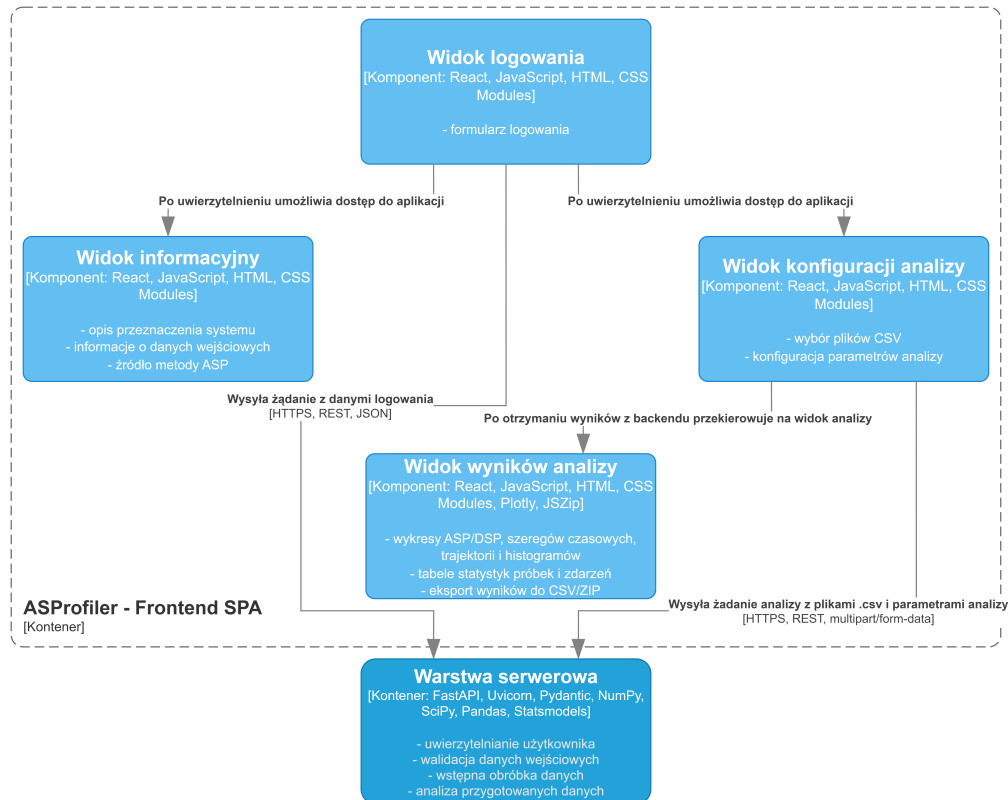
Komunikacja pomiędzy warstwami odbywa się za pomocą protokołu HTTPS. Pliki CSV oraz parametry analizy przesyłane są do serwera w ramach pojedynczego żądania multipart/form-data, a wyniki obliczeń zwracane są w formacie JSON.

Warstwa kliencka odpowiada za import plików, konfigurację parametrów analizy, filtrowanie i agregację danych oraz prezentację wyników i ich eksport. Warstwa serwerowa uwierzytelnia użytkownika, waliduje dane wejściowe i wykonuje obliczenia analityczne.

System nie wykorzystuje bazy danych do trwałego przechowywania danych treningowych ani wyników analiz. Każde żądanie zawiera komplet informacji niezbędnych do wykonania obliczeń, a uzyskane wyniki są zwracane do aplikacji klienckiej i przechowywane w jej stanie.

#### 4.1.1. Komponenty warstwy klienckiej

Komponenty warstwy klienckiej przedstawiono na rysunku 4.2.



**Rysunek 4.2.** Diagram C3 komponentów warstwy klienckiej.

Interfejs użytkownika podzielono na cztery niezależne widoki odpowiadające kolejnym etapom pracy z systemem.

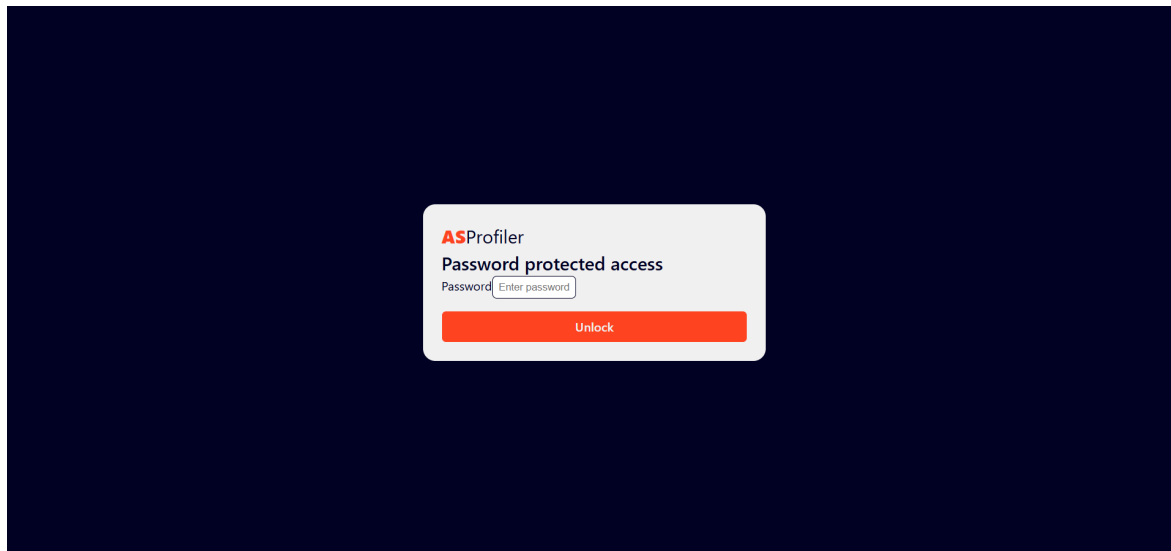
Widok logowania, przedstawiony na rysunku 4.3, umożliwia uwierzytelnienie użytkownika i uzyskanie dostępu do pozostałych widoków aplikacji.

Widok informacyjny, przedstawiony na rysunku 4.4, zawiera informacje o przeznaczeniu aplikacji, obsługiwanych formatach plików wejściowych oraz źródle metody ASP.

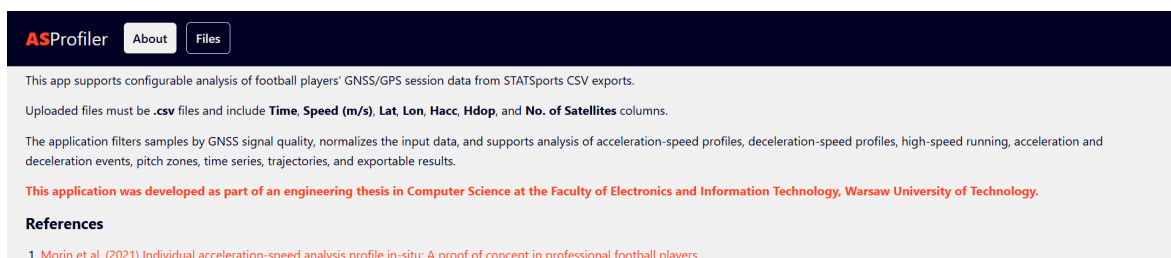
Widok listy plików odpowiada za wybór danych wejściowych oraz konfigurację analizy przed przesłaniem danych do warstwy serwerowej.

Użytkownik może dodać jeden lub wiele plików CSV poprzez wybór z dysku lub przeciągnięcie ich do wyznaczonego obszaru interfejsu. Formularz konfiguracji parametrów analizy, przedstawiony na rysunku 4.5, umożliwia określenie domyślnych parametrów analizy. Parametry jakości sygnału GNSS, filtrowania prędkości oraz wyznaczania profili ASP i DSP, takie jak szerokość przedziałów prędkości, liczba punktów wybieranych z każdego przedziału i poziom ufności wykorzystywany przy odrzucaniu punktów odstających, mają charakter globalny i nie są nadpisywane dla poszczególnych plików.

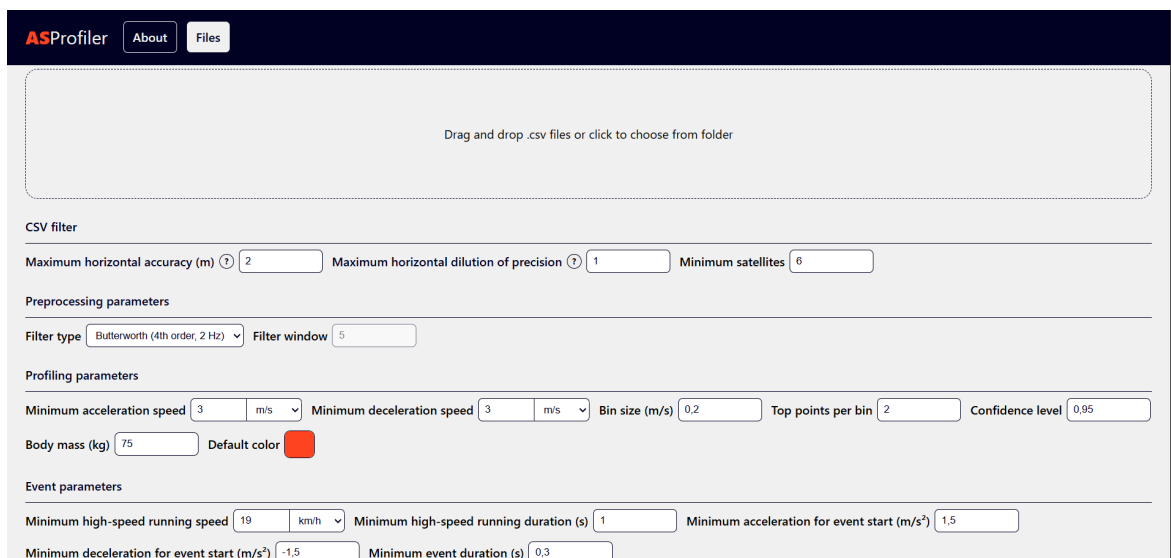
## 4. Projekt i implementacja systemu



Rysunek 4.3. Widok logowania.



Rysunek 4.4. Widok informacyjny.



Rysunek 4.5. Widok listy plików. Dodawanie plików i formularz konfiguracji globalnych parametrów analizy.

Tabela plików wejściowych, przedstawiona na rysunku 4.6, umożliwia nadpisywanie pozostałych parametrów dla poszczególnych plików. Konfiguracja i preferencje użytkownika są przechowywane lokalnie w przeglądarce z wykorzystaniem mechanizmu `localStorage`.

The screenshot shows a web interface for managing analysis files. At the top, there are buttons for 'Select all', 'Analyze selected', and 'Remove selected'. Below this is a table with columns for 'File name' and 'Size'. Two files are listed: '2026-09-10-Player-01.csv' (81.0 MB) and '2026-09-10-Player-02.csv' (93.3 MB). Each file has an 'Analyze' and 'Remove' button. Below the file list, there are input fields for various parameters for each file, including Color, Body mass (kg), Minimum acceleration speed, Minimum deceleration speed, Minimum high-speed running speed, Minimum high-speed running duration (s), Minimum acceleration for event start (m/s²), Minimum deceleration for event start (m/s²), and Minimum event duration (s).

**Rysunek 4.6.** Widok listy plików. Wybór i nadpisywanie parametrów analizy dla poszczególnych plików.

Widok analizy składa się z sekcji profili ASP, DSP i FVP, sekcji statystyk próbek, sekcji szeregów czasowych i trajektorii oraz sekcji histogramów i statystyk zdarzeń.

The screenshot shows the 'Analysis mode' dropdown set to 'Acceleration / Deceleration'. Below it is a table with columns: File name, Analysis Profile, Analysis Profile Equation,  $A_0$ ,  $V_0$ ,  $R^2$ , Body mass, and Color. There are four rows of data, each with a checkbox in the first column. The first two rows are for 'Acceleration' and the last two for 'Deceleration'.

|                                     | File name                | Analysis Profile | Analysis Profile Equation     | $A_0$                  | $V_0$                 | $R^2$ | Body mass | Color     |
|-------------------------------------|--------------------------|------------------|-------------------------------|------------------------|-----------------------|-------|-----------|-----------|
| <input checked="" type="checkbox"/> | 2026-09-10-Player-01.csv | Acceleration     | $a = 7.21 + (-0.74) \cdot v$  | 7.21 m/s <sup>2</sup>  | 9.74 m/s (35.08 km/h) | 0.99  | 75.00 kg  | Orange    |
| <input checked="" type="checkbox"/> | 2026-09-10-Player-02.csv | Acceleration     | $a = 9.02 + (-0.98) \cdot v$  | 9.02 m/s <sup>2</sup>  | 9.24 m/s (33.25 km/h) | 0.99  | 80.00 kg  | Dark Blue |
| <input checked="" type="checkbox"/> | 2026-09-10-Player-01.csv | Deceleration     | $a = 11.26 + (-1.20) \cdot v$ | 11.26 m/s <sup>2</sup> | 9.41 m/s (33.87 km/h) | 0.99  | 75.00 kg  | Orange    |
| <input checked="" type="checkbox"/> | 2026-09-10-Player-02.csv | Deceleration     | $a = 13.29 + (-1.39) \cdot v$ | 13.29 m/s <sup>2</sup> | 9.59 m/s (34.51 km/h) | 0.99  | 80.00 kg  | Dark Blue |

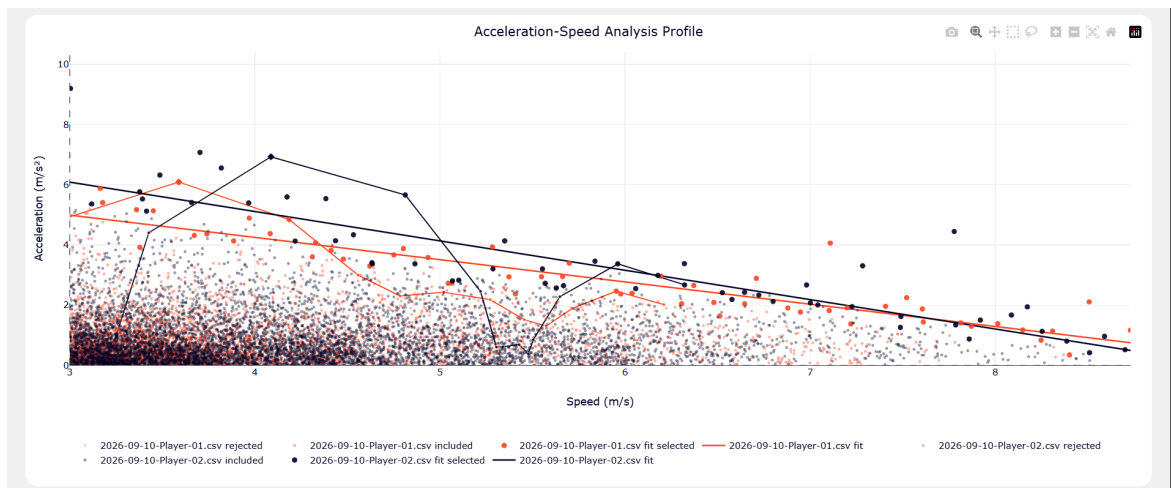
**Rysunek 4.7.** Widok analizy. Parametry profili i ustawienia widoku.

Tabela, przedstawiona na rysunku 4.7, umożliwia zmianę koloru wyników poszczególnych plików oraz ukrywanie wybranych serii przyspieszeń i hamowań. Dla każdego pliku prezentowane są parametry dopasowania profili, obejmujące teoretyczne maksymalne przyspieszenie  $A_0$ , teoretyczną maksymalną prędkość  $V_0$ , równanie regresji oraz współczynnik determinacji  $R^2$ . W tabelach wartości prędkości prezentowane są równolegle w m/s i km/h.

Użytkownik może wybrać kinematyczny lub mechaniczny tryb prezentacji wyników profili, tabeli ogólnych statystyk oraz wykresu szeregu czasowego. Widok umożliwia eksport pełnego pakietu wyników i osobne eksporty danych z tabeli profili, statystyk oraz zdarzeń.

Wykres ASP, przedstawiony na rysunku 4.8, prezentuje próbki oznaczone kolorami przypisanymi do poszczególnych plików. Większymi markerami wyróżniono punkty uwzględnione w końcowej regresji. Analogiczny sposób prezentacji zastosowano dla wykresów DSP oraz odpowiadających im wykresów FVP.

## 4. Projekt i implementacja systemu



**Rysunek 4.8.** Widok analizy. Wykres profilu ASP z próbkami zaznaczonymi przez użytkownika i wyróżnionymi punktami sąsiednimi.

Użytkownik może zaznaczać punkty na wykresach profili. Zaznaczone punkty są jednocześnie wyróżniane na szeregach czasowych prędkości i przyspieszenia oraz na trajektorii zawodnika, co umożliwia analizę ich kontekstu czasowego i przestrzennego.



**Rysunek 4.9.** Widok analizy. Konfiguracja liczby punktów wyróżnianych wokół zaznaczonej próbki i zarządzanie zaznaczonymi próbkami w pasku bocznym.

Pasek boczny, przedstawiony na rysunku 4.9, umożliwia określenie liczby punktów wyróżnianych przed i po zaznaczonej próbce oraz przeglądanie, sortowanie i usuwanie wybranych próbek.

Tabela ogólnych statystyk, przedstawiona na rysunku 4.10, prezentuje dla prędkości, przyspieszenia, hamowania oraz odpowiadających im sił wartości minimalne, średnie, mediany i maksima, a także czas trwania i pole pod krzywą. Pole pod krzywą odpowiada dystansowi dla prędkości, łącznej zmianie prędkości dla przyspieszenia i hamowania oraz impulsowi dla siły. Przyspieszenia i hamowania analizowane są oddzielnie, co pozwala

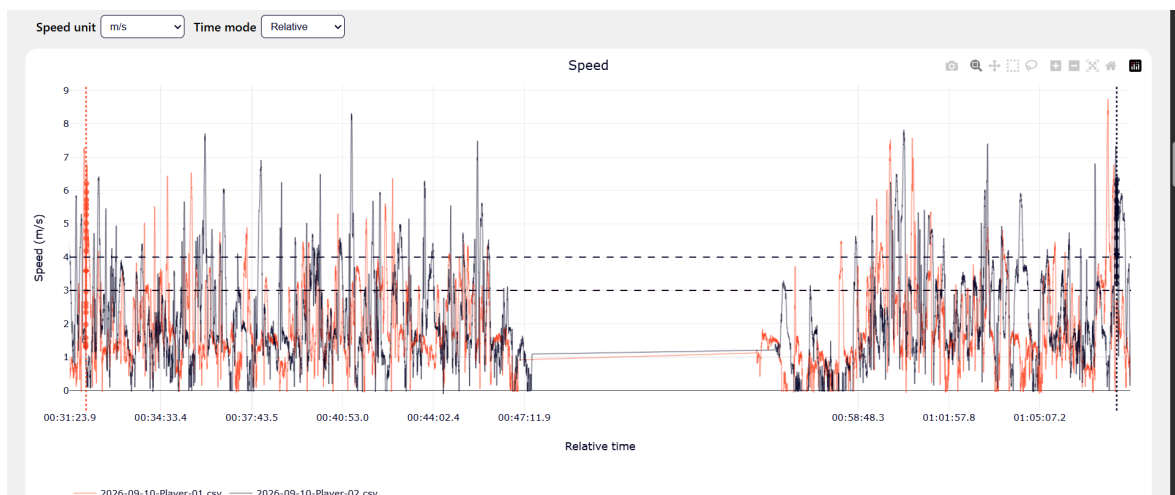
Activity scope: All Pitch zone: Full pitch

Filtered GPS data stats Export CSV

|                                       | Min                       | Mean                    | Median                  | Max                      | Duration ? | Area ?      |
|---------------------------------------|---------------------------|-------------------------|-------------------------|--------------------------|------------|-------------|
| 2026-09-10-Player-01.csv Speed        | -0.08 m/s<br>(-0.28 km/h) | 1.83 m/s<br>(6.60 km/h) | 1.51 m/s<br>(5.43 km/h) | 8.74 m/s<br>(31.47 km/h) | 00:41:16.8 | 8.67 km     |
| 2026-09-10-Player-02.csv Speed        | -0.11 m/s<br>(-0.40 km/h) | 2.11 m/s<br>(7.59 km/h) | 1.75 m/s<br>(6.31 km/h) | 8.70 m/s<br>(31.33 km/h) | 00:47:32.5 | 11.55 km    |
| 2026-09-10-Player-01.csv Acceleration | 0.00 m/s <sup>2</sup>     | 0.73 m/s <sup>2</sup>   | 0.49 m/s <sup>2</sup>   | 9.60 m/s <sup>2</sup>    | 00:41:16.8 | 2558.03 m/s |
| 2026-09-10-Player-02.csv Acceleration | 0.00 m/s <sup>2</sup>     | 0.79 m/s <sup>2</sup>   | 0.53 m/s <sup>2</sup>   | 10.43 m/s <sup>2</sup>   | 00:47:32.5 | 2598.78 m/s |
| 2026-09-10-Player-01.csv Deceleration | 0.00 m/s <sup>2</sup>     | 0.74 m/s <sup>2</sup>   | 0.48 m/s <sup>2</sup>   | 9.02 m/s <sup>2</sup>    | 00:41:16.8 | 1633.00 m/s |
| 2026-09-10-Player-02.csv Deceleration | 0.00 m/s <sup>2</sup>     | 0.82 m/s <sup>2</sup>   | 0.53 m/s <sup>2</sup>   | 10.74 m/s <sup>2</sup>   | 00:47:32.5 | 2090.71 m/s |

**Rysunek 4.10.** Widok analizy. Tabela ogólnych statystyk.

osobno porównywać ich charakterystyki. Użytkownik może ograniczyć zakres statystyk do wybranej strefy boiska lub odcinków HSR.



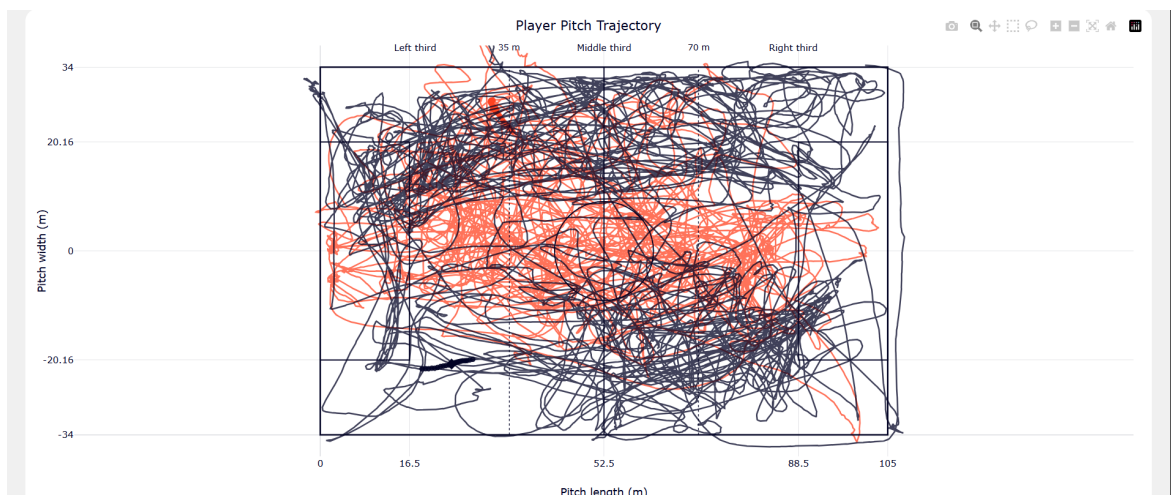
**Rysunek 4.11.** Widok analizy. Szereg czasowy prędkości z wyróżnionymi punktami.

Szereg czasowy prędkości, przedstawiony na rysunku 4.11, prezentuje zmiany prędkości w czasie. Analogiczny wykres jest dostępny dla przyspieszenia lub odpowiadającej mu siły. Użytkownik może wybrać jednostkę prędkości, m/s lub km/h. Może też wybrać względną skalę osi czasu, wyrażoną w sekundach od początku sesji, lub skalę bezwzględną, zgodną ze znacznikami czasu w pliku źródłowym. Na szeregu czasowym prędkości zaznaczone są minimalne prędkości wykorzystywane przy wyznaczaniu profili ASP i DSP.

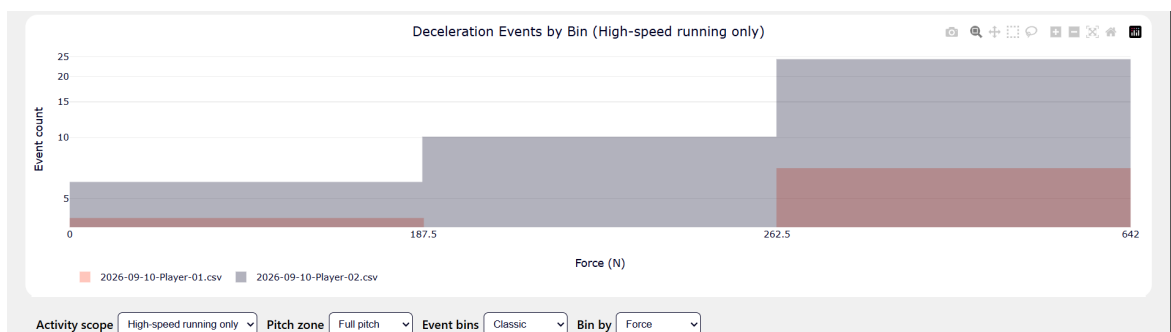
Trajektoria, przedstawiona na rysunku 4.12, prezentuje przebieg ruchu zawodnika na tle standardowego boiska piłkarskiego o wymiarach 105 na 68 m, z zaznaczonym podziałem na strefę lewą, środkową i prawą.

Histogram zdarzeń hamowania, przedstawiony na rysunku 4.13, prezentuje rozkład zdarzeń przypisanych do przedziałów na podstawie ich wartości maksymalnych. Analogiczny sposób prezentacji zastosowano dla zdarzeń przyspieszenia. Dla próbek prędkości dostępny jest histogram z przedziałami o szerokości 1 m/s. Użytkownik może wybrać klasyczny podział zdarzeń na przedziały 0–2,5, 2,5–3,5 i powyżej 3,5 m/s<sup>2</sup> lub schemat

## 4. Projekt i implementacja systemu



**Rysunek 4.12.** Widok analizy. Trajektoria zawodnika na tle boiska z wyróżnionymi punktami.



**Rysunek 4.13.** Widok analizy. Histogram zdarzeń hamowania.

szczegółowy z przedziałami 0–3, 3–4, 4–5, 5–6, 6–7 oraz powyżej 7 m/s<sup>2</sup>. Może też wybrać kryterium podziału według wartości przyspieszenia lub siły poziomej. Krawędzie przedziałów siłowych odpowiadają progom przyspieszenia przeliczonym dla standardowej masy 75 kg. Użytkownik może również wybrać liniową lub logarytmiczną skalę osi licznosci oraz ograniczyć zakres prezentowanych danych do wybranej strefy boiska lub odcinków HSR. Wybrane ustawienia widoku wpływają również na wyniki w tabelach zdarzeń.

| Acceleration event early/late phase statistics <sup>?</sup> |                          |               |               |                           |                        |                                       |                                    |                       |
|-------------------------------------------------------------|--------------------------|---------------|---------------|---------------------------|------------------------|---------------------------------------|------------------------------------|-----------------------|
| File name                                                   | Bin (m/s <sup>2</sup> )  | Mean duration | Mean distance | Mean acceleration average | Mean acceleration peak | Mean relative PP average <sup>?</sup> | Mean relative PP peak <sup>?</sup> | Mean HPI <sup>?</sup> |
| 2026-09-10-Player-01.csv                                    | All early                | 0.45 s        | 2.67 m        | 1.64 m/s <sup>2</sup>     | 2.33 m/s <sup>2</sup>  | 9.70 W/kg                             | 13.67 W/kg                         | 0.69 N-s              |
|                                                             | All late                 | 0.57 s        | 3.83 m        | 1.07 m/s <sup>2</sup>     | 1.72 m/s <sup>2</sup>  | 7.04 W/kg                             | 11.08 W/kg                         | 0.54 N-s              |
|                                                             | All early/late           | 0.72          | 0.70          | 1.53                      | 1.35                   | 1.38                                  | 1.23                               | 1.28                  |
| 2026-09-10-Player-02.csv                                    | All early                | 0.46 s        | 2.68 m        | 1.93 m/s <sup>2</sup>     | 2.58 m/s <sup>2</sup>  | 10.96 W/kg                            | 14.42 W/kg                         | 0.83 N-s              |
|                                                             | All late                 | 0.64 s        | 4.31 m        | 1.16 m/s <sup>2</sup>     | 1.92 m/s <sup>2</sup>  | 7.43 W/kg                             | 12.05 W/kg                         | 0.65 N-s              |
|                                                             | All early/late           | 0.72          | 0.62          | 1.66                      | 1.35                   | 1.47                                  | 1.20                               | 1.28                  |
| 2026-09-10-Player-01.csv                                    | Low (0-187.5) early      | 0.45 s        | 2.64 m        | 1.52 m/s <sup>2</sup>     | 2.05 m/s <sup>2</sup>  | 8.90 W/kg                             | 11.76 W/kg                         | 0.65 N-s              |
|                                                             | Low (0-187.5) late       | 0.55 s        | 3.67 m        | 0.92 m/s <sup>2</sup>     | 1.53 m/s <sup>2</sup>  | 5.93 W/kg                             | 9.57 W/kg                          | 0.49 N-s              |
|                                                             | Low (0-187.5) early/late | 0.81          | 0.72          | 1.65                      | 1.34                   | 1.50                                  | 1.23                               | 1.31                  |
| 2026-09-10-Player-02.csv                                    | Low (0-187.5) early      | 0.51 s        | 2.98 m        | 1.49 m/s <sup>2</sup>     | 2.15 m/s <sup>2</sup>  | 8.62 W/kg                             | 11.85 W/kg                         | 0.72 N-s              |
|                                                             | Low (0-187.5) late       | 0.63 s        | 4.20 m        | 0.93 m/s <sup>2</sup>     | 1.52 m/s <sup>2</sup>  | 5.96 W/kg                             | 9.74 W/kg                          | 0.59 N-s              |
|                                                             | Low (0-187.5) early/late | 0.81          | 0.71          | 1.60                      | 1.42                   | 1.45                                  | 1.22                               | 1.21                  |

**Rysunek 4.14.** Widok analizy. Statystyki wykrytych zdarzeń przyspieszenia z podziałem na fazy.

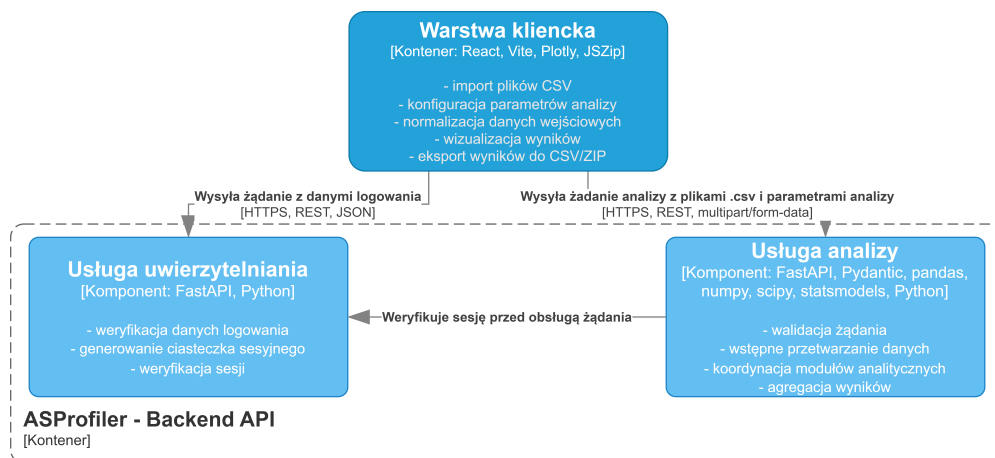
Tabela statystyk zdarzeń przyspieszenia, przedstawiona na rysunku 4.14, prezentuje statystyki zagregowane w przedziałach intensywności. Analogiczny sposób prezentacji zastosowano dla zdarzeń hamowania. Użytkownik może wybrać prezentację statystyk całych zdarzeń albo ich faz wczesnych i późnych.

W przypadku braku podziału na fazy tabela prezentuje liczbę i gęstość zdarzeń, średni czas trwania i dystans, średnie przyspieszenie lub hamowanie, średnią prędkość wejściową i wyjściową, średnią moc poziomą w przeliczeniu na kilogram masy ciała, średni poziomy impuls oraz stosunek liczby zdarzeń do zdarzeń przeciwnego rodzaju.

W przypadku podziału na fazy dla każdego przedziału prezentowane są uśrednione wartości czasu trwania, dystansu, średniego i maksymalnego przyspieszenia lub hamowania, średniej i maksymalnej mocy oraz impulsu. Granicę pomiędzy fazą wczesną i późną wyznacza pierwsza próbka, w której prędkość osiąga wartość w połowie między prędkością wejścia i wyjścia ze zdarzenia. Kryterium to różni się od spotykanego w literaturze podziału względem 50% prędkości początkowej hamowania. Zmiana wynika z obserwacji, że w warunkach meczowych zawodnik rzadko wytraca prędkość do zera, przez co próg 50% prędkości początkowej często nie jest osiągnięty.

#### 4.1.2. Komponenty warstwy serwerowej

Komponenty warstwy serwerowej przedstawiono na rysunku 4.15.



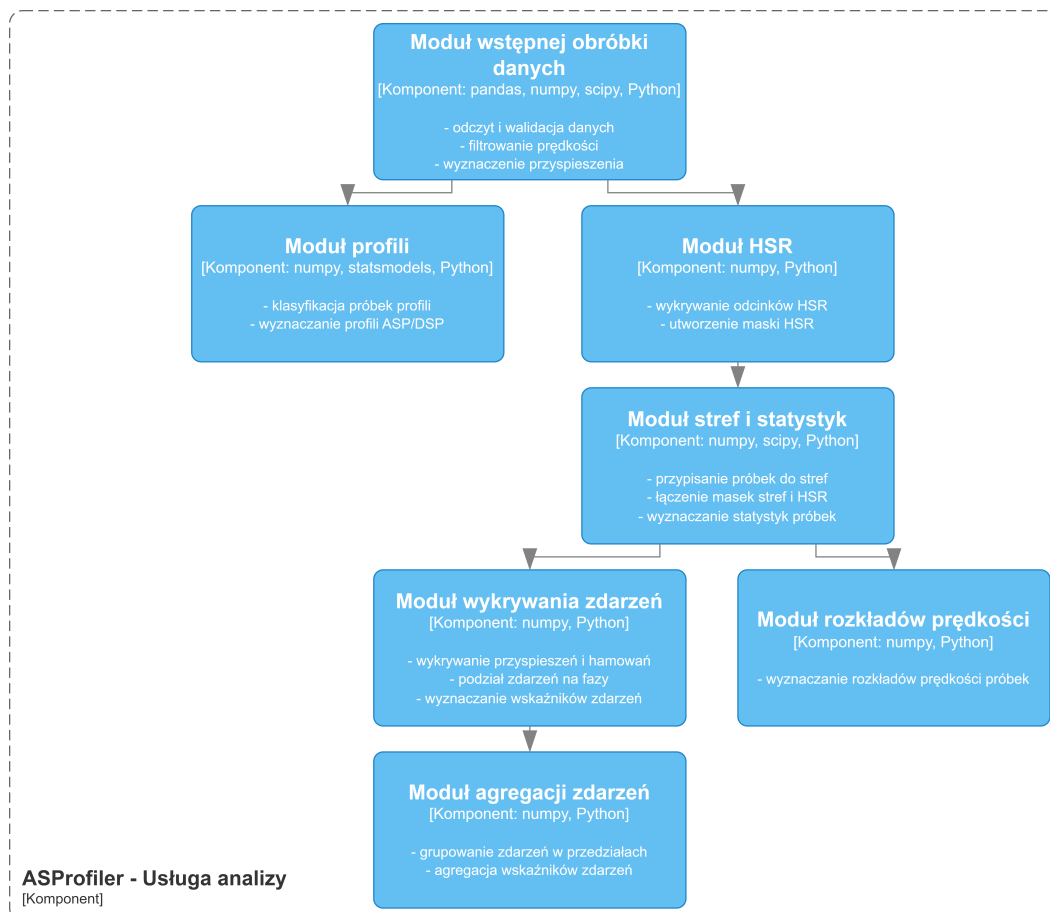
**Rysunek 4.15.** Diagram C3 komponentów warstwy serwerowej.

Obejmują one usługę uwierzytelniania oraz usługę analizy. Usługa uwierzytelniania odpowiada za logowanie użytkownika, utworzenie podpisanego ciasteczka sesyjnego, weryfikację jego ważności oraz kontrolę dostępu do chronionych funkcji analitycznych.

Strukturę usługi analizy przedstawiono na rysunku 4.16.

Koordynuje ona pracę siedmiu wyspecjalizowanych modułów analitycznych. Odpowiada za walidację żądania, koordynację modułów oraz agregację wyników do jednolitej struktury odpowiedzi.

Wstępna obróbka obejmuje przekształcenie przesłanego pliku CSV do wewnętrznej reprezentacji, filtrowanie sygnału prędkości oraz wyznaczenie przyspieszenia na podstawie jego pochodnej numerycznej. Otrzymany szereg czasowy stanowi wspólne wejście dla



Rysunek 4.16. Diagram modułów usługi analizy.

kolejnych modułów analitycznych. Na jego podstawie wyznaczane są profile ASP i DSP, wykrywane są odcinki HSR oraz przypisywane są próbki do stref boiska. Informacje o strefach i odcinkach HSR są następnie wykorzystywane przy obliczaniu statystyk próbek, wykrywaniu zdarzeń przyspieszenia i hamowania, agregacji wskaźników zdarzeń oraz wyznaczaniu rozkładów prędkości.

## 4.2. Model danych

Model danych opisuje struktury danych przekazywane w systemie, czyli dane wejściowe, strukturę zapytania przekazywanego do warstwy serwerowej oraz format wyników analizy. Ze względu na brak trwałego przechowywania nie ma on postaci klasycznego modelu encji charakterystycznego dla aplikacji bazodanowych.

### 4.2.1. Dane wejściowe

Przykładowy fragment pliku wejściowego przedstawiono na listingu 1.

**Listing 1.** Fragment pliku CSV eksportowanego z platformy STATSports.

```
Player Display Name,Time,Lat, Lon,Speed (m/s),Heart Rate (bpm),Hacc  
Hdop,Quality of Signal,No. of Satellites,Instantaneous Acceleration  
Impulse,Accl X,Accl Y,Accl Z,Gyro X,Gyro Y,Gyro Z
```

```
"Player 01",00:00:36.0,49.9997672,0.0000742,0.947,82,4,1.2,0,0,0.444,
-0.2569,0.6559,0.6888,-11.90,-61.95,0.77
"Player 01",00:00:36.0,49.9997672,0.0000742,0.947,82,4,1.2,0,0,0.444,
-0.3638,0.5885,0.6076,-20.09,-53.97,2.73
...
"Player 01",00:00:36.1,49.9997680,0.0000755,1.097,82,4,1.2,347,22,
1.500,-0.2064,0.8960,0.4553,-22.26,-2.03,29.33
```

Danymi wejściowymi systemu są pliki CSV eksportowane z platformy STATSports. Pliki zawierają dane GNSS rejestrowane z częstotliwością 10 Hz oraz dane z IMU rejestrowane z częstotliwością 100 Hz. W rezultacie dla dziesięciu kolejnych rekordów o tym samym znaczniku czasu wartości położenia i prędkości pozostają niezmiennione, natomiast zmieniają się wartości z IMU.

Analiza wykorzystuje czas pomiaru, współrzędne geograficzne zawodnika oraz chwilową prędkość. Przyspieszenie jest wyznaczane w warstwie serwerowej na podstawie przefiltrowanego sygnału prędkości. Z plików odczytywane są również wskaźniki jakości sygnału, czyli dokładność pozioma (Hacc, ang. *Horizontal Accuracy*), współczynnik HDOP oraz liczba widocznych satelitów, wykorzystywane do odrzucania próbek niespełniających przyjętych kryteriów jakości.

Dane dotyczące tętna pominięto ze względu na liczne wartości równe zero w dostępnych plikach źródłowych. Pozostałe dane, w tym pomiary z akcelerometru i żyroskopu, nie są wykorzystywane przez zaimplementowane metody analityczne.

#### 4.2.2. Kontrakt wymiany danych klient-serwer

Kontrakt wymiany danych między warstwą kliencką a serwerową obejmuje parametry konfiguracji analizy oraz znormalizowane pliki z szeregami czasowymi, przygotowane po stronie klienta.

Strukturę obiektu JSON z parametrami analizy przedstawiono na listingu 2.

**Listing 2.** Obiekt JSON z parametrami analizy przekazywany do warstwy serwerowej.

```
{
  "per_file_params": {
    "Player_01.csv": {
      "min_acc_speed_m_per_s": 3.5,
      "min_dec_speed_m_per_s": 3.0,
      "bin_size_m_per_s": 0.2,
      "extreme_count": 2,
      "confidence_level": 0.95,
      "body_mass_kg": 87.0,
      "min_high_speed_running_speed_m_per_s": 5.28,
      "min_high_speed_running_duration_s": 1.0,
      "min_acc_start_m_per_s2": 1.5,
      "min_dec_start_m_per_s2": -1.5,
```

```

        "min_event_duration_s": 0.3
    },
    "preprocessing_params": {
        "filter_mode": "butterworth",
        "filter_window_samples": 5
    }
}

```

Parametry analizy przekazywane są osobno dla każdego pliku, natomiast parametry filtrowania sygnału prędkości są wspólne dla całego żądania. Parametry jakości sygnału GNSS są wykorzystywane wcześniej, podczas przygotowania danych w warstwie klienckiej.

Zestaw parametrów konfiguracji wykorzystywanych przez system przedstawiono w tabeli 4.1.

**Tabela 4.1.** Parametry konfiguracji wykorzystywane przez system.

| Parametr                             | Domyślna wartość      | Dopuszczalne wartości                        |
|--------------------------------------|-----------------------|----------------------------------------------|
| min_acc_speed_m_per_s                | 3,0 m/s               | $\geq 0$                                     |
| min_dec_speed_m_per_s                | 3,0 m/s               | $\geq 0$                                     |
| bin_size_m_per_s                     | 0,2 m/s               | $> 0$                                        |
| extreme_count                        | 2                     | $\geq 1$                                     |
| confidence_level                     | 0,95                  | $0 < x < 1$                                  |
| body_mass_kg                         | 75 kg                 | $> 0$                                        |
| min_high_speed_running_speed_m_per_s | 19 km/h<br>(5,28 m/s) | $> 0$                                        |
| min_high_speed_running_duration_s    | 1,0 s                 | $> 0$                                        |
| min_acc_start_m_per_s2               | 1,5 m/s <sup>2</sup>  | $> 0$                                        |
| min_dec_start_m_per_s2               | -1,5 m/s <sup>2</sup> | $< 0$                                        |
| min_event_duration_s                 | 0,3 s                 | $> 0$                                        |
| filter_mode                          | butterworth           | none, median, mean, median_mean, butterworth |
| filter_window_samples                | 5                     | $\geq 1$                                     |
| maxHorizontalAccuracyM               | 2 m                   | $\geq 0$                                     |
| maxHorizontalDilutionOfPrecision     | 1                     | $\geq 0$                                     |
| minSatelliteCount                    | 6                     | $\geq 0$                                     |

Parametry przedstawione w tabeli obejmują konfigurację jakości sygnału GNSS, filtrowania prędkości, wyznaczania profili ASP i DSP, wykrywania zdarzeń oraz HSR. Dla profili ASP i DSP określają minimalną prędkość próbek, szerokość przedziałów prędkości, liczbę punktów skrajnych wybieranych z każdego przedziału i poziom ufności, dla zdarzeń progi rozpoczęcia przyspieszenia i hamowania oraz minimalny czas trwania, a dla HSR

minimalną prędkość i minimalny czas trwania odcinka. Wartości domyślne dobrano na podstawie przeglądu rozwiązań komercyjnych i literatury oraz konsultacji z ekspertem z klubu sportowego.

Strukturę znormalizowanego pliku CSV przekazywanego do warstwy serwerowej przedstawiono na listingu 3.

**Listing 3.** Fragment znormalizowanego pliku CSV przekazywanego do warstwy serwerowej.

```
time , speed , lat , lon
10:00:50.9,0,50.7775353,19.160678
10:00:51.0,0,50.7775353,19.160678
```

W warstwie klienckiej odrzucane są rekordy niespełniające przyjętych kryteriów jakości sygnału GNSS, określonych przez maksymalne wartości Hacc i HDOP oraz minimalną liczbę widocznych satelitów. Pozostałe rekordy są następnie grupowane według znacznika czasu, a wartości prędkości i współrzędnych geograficznych uśredniane w obrębie każdej grupy. W wyniku tego przetwarzania dla każdego zachowanego znacznika czasu powstaje pojedynczy rekord zawierający czas, prędkość oraz współrzędne geograficzne zawodnika.

#### 4.2.3. Wyjście systemu

Uproszczoną strukturę odpowiedzi interfejsu API przedstawiono na listingu 4.

**Listing 4.** Uproszczona struktura odpowiedzi interfejsu API.

```
{
  "results": [
    {
      "file_name": "Player_01.csv",
      "analysis": {
        "acc_profile_fit": { ... },
        "dec_profile_fit": { ... },
        "acc_events": { ... },
        "dec_events": { ... },
        "time_series": { ... },
        "sample_stats": { ... },
        "sample_distributions": { ... },
        "meta": { ... },
      }
    },
    {
      "file_name": "Player_02.csv",
      "error": "CSV time values must be strictly increasing"
    }
  ]
}
```

Każdy element odpowiedzi odpowiada jednemu przesłanemu plikowi. Dla poprawnie przetworzonych plików zwracane są dane potrzebne do prezentacji wyników analizy, obejmujące parametry dopasowania profili ASP i DSP, wykryte zdarzenia przyspieszeń i hamowań, szereg czasowy, statystyki próbek, rozkłady prędkości oraz użyte parametry analizy. W przypadku błędu zwracana jest nazwa pliku wraz z komunikatem diagnostycznym, a wyniki pozostałych poprawnie przetworzonych plików pozostają dostępne.

Część wielkości mechanicznych prezentowanych w interfejsie jest wyznaczana w warstwie klienckiej na podstawie wyników zwróconych przez warstwę serwerową oraz masy zawodnika podanej w konfiguracji. Dotyczy to prostych przeliczeń, takich jak wyznaczanie siły na podstawie przyspieszenia, mocy bezwzględnej na podstawie mocy względnej oraz impulsu siły na podstawie zmiany prędkości.

Na podstawie odpowiedzi interfejsu API aplikacja kliencka umożliwia eksport wyników analizy do plików CSV lub archiwów ZIP. Archiwum zawiera parametry dopasowania profili ASP i DSP z przefiltrowanymi punktami szeregu, listy wykrytych zdarzeń przyspieszeń i hamowań z zagregowanymi statystykami, statystyki próbek oraz parametry analizy. Pliki z punktami szeregu czasowego zawierają informację o ich klasyfikacji jako odrzuconych, kandydujących lub użytych w końcowej regresji profilu. Pliki zdarzeń i statystyk zachowują metryki kinematyczne i mechaniczne wraz z zakresem analizy, strefą boiska, schematem i kryterium podziału na przedziały oraz, w przypadku zdarzeń, podziałem na fazę wczesną i późną. Umożliwia to dalszą analizę danych poza aplikacją.

### 4.3. Zastosowane technologie

Do implementacji systemu wykorzystano technologie umożliwiające budowę interaktywnego interfejsu użytkownika, udostępnienie interfejsu REST API, przetwarzanie danych numerycznych oraz wdrożenie aplikacji w środowisku chmurowym.

#### 4.3.1. React i biblioteki warstwy klienckiej

Warstwę kliencką zaimplementowano w języku JavaScript z wykorzystaniem biblioteki React. Wybór tej technologii wynikał z potrzeby budowy interaktywnego interfejsu użytkownika obejmującego formularze konfiguracji analizy, wykresy oraz dynamicznie aktualizowane tabele wyników.

Jako środowisko deweloperskie wykorzystano Vite. W procesie budowania wersji produkcyjnej narzędzie to generuje pliki statyczne, które mogą być następnie udostępniane bez konieczności utrzymywania osobnego serwera aplikacyjnego dla warstwy klienckiej.

Do interaktywnych wizualizacji wyników wykorzystano bibliotekę Plotly, zapewniającą obsługę wyboru punktów, przybliżanie fragmentów wykresu oraz ukrywanie i wyświetlanie poszczególnych serii.

Do eksportu wyników wykorzystano bibliotekę JSZip, umożliwiającą tworzenie archiwów ZIP bezpośrednio w przeglądarce.

#### 4.3.2. Python i biblioteki analityczne

Moduły analityczne zaimplementowano w języku Python, co pozwoliło wykorzystać sprawdzone biblioteki do analizy danych, obliczeń numerycznych i przetwarzania sygnałów.

Do obliczeń numerycznych wykorzystano bibliotekę NumPy, zapewniającą operacje wektorowe na szeregach czasowych i tablicach danych. Biblioteka Pandas posłużyła do przetwarzania danych tabelarycznych importowanych z plików CSV, natomiast SciPy dostarczyła funkcji wykorzystywanych do filtrowania metodą Butterwortha oraz wyznaczania obwiedni podczas analizy orientacji boiska. Do wyznaczania przedziałów ufności regresji liniowej wykorzystano bibliotekę Statsmodels.

Zastosowanie tych bibliotek pozwoliło na skupienie się na implementacji metod analizy danych GNSS, a nie podstawowych operacji numerycznych.

#### 4.3.3. FastAPI i Pydantic

Warstwę API zaimplementowano z wykorzystaniem FastAPI. Technologię tę wybrano, ponieważ pozwala na łatwe budowanie interfejsów API w języku Python i integruje się z biblioteką Pydantic.

Bibliotekę Pydantic wykorzystano do definiowania modeli danych, walidacji struktury i typów parametrów wejściowych oraz konwersji danych pomiędzy formatem JSON a obiektami języka Python na podstawie adnotacji typów.

Uvicorn pełni rolę serwera ASGI obsługującego aplikację FastAPI.

#### 4.3.4. Amazon Web Services

System wdrożono w środowisku Amazon Web Services (AWS). Wykorzystanie AWS pozwoliło utrzymać warstwę kliencką i serwerową oraz obsłużyć domenę i certyfikaty TLS w ramach jednej platformy chmurowej. Istotnym kryterium wyboru była również możliwość ograniczenia kosztów infrastruktury dzięki darmowemu planowi wybranych usług.

Warstwę kliencką udostępniono jako aplikację statyczną za pomocą usługi Amazon S3 do przechowywania plików statycznych oraz sieci dystrybucji treści CloudFront. Warstwę serwerową uruchomiono na instancji Amazon EC2 z systemem Ubuntu, na której aplikacja FastAPI działa jako usługa systemowa obsługiwana przez serwer ASGI Uvicorn. Ruch przychodzący do warstwy serwerowej jest przekazywany przez Nginx pełniący funkcję serwera pośredniczącego.

Do obsługi domeny wykorzystano Route 53. Certyfikat warstwy klienckiej jest zarządzany przez AWS Certificate Manager. Certyfikat warstwy serwerowej jest wystawiany przez Let's Encrypt, a jego obsługę i odnawianie na instancji EC2 realizuje narzędzie certbot. Osobne certyfikaty wynikają z udostępniania warstwy klienckiej i serwerowej pod różnymi nazwami hostów.

### 4.4. Implementacja algorytmów analitycznych

#### 4.4.1. Filtr Butterwortha

Domyślną metodą wygładzania sygnału prędkości jest dolnoprzepustowy filtr Butterwortha. Implementację przedstawiono na listingu 5.

**Listing 5.** Implementacja filtra Butterwortha dla sygnału prędkości.

```
def _apply_butterworth_low_pass(
    values: np.ndarray, times: np.ndarray) -> np.ndarray:
    padding_length = 3 * (_BUTTERWORTH_ORDER + 1)
    if len(values) <= padding_length:
        return values

    sample_rate = 1 / float(np.median(np.diff(times)))
    nyquist_frequency = sample_rate / 2
    if _BUTTERWORTH_CUTOFF_FREQUENCY >= nyquist_frequency:
        return values

    b_coefs, a_coefs = butter(
        _BUTTERWORTH_ORDER,
        _BUTTERWORTH_CUTOFF_FREQUENCY / nyquist_frequency,
        btype="low")

    return filtfilt(
        b_coefs, a_coefs, values, padlen=padding_length)
```

Zastosowano filtr Butterwortha drugiego rzędu z częstotliwością odcięcia równą 2 Hz. Filtrowanie realizowane jest dwukierunkowo za pomocą funkcji `scipy.signal.filtfilt`, dzięki czemu nie wprowadza przesunięcia fazowego, a efektywny rząd filtru jest podwojony.

Częstotliwość próbkowania wyznaczana jest na podstawie mediany odstępów czasowych pomiędzy kolejnymi próbkami, co ogranicza wpływ brakujących próbek. Na jej podstawie obliczana jest częstotliwość Nyquista, równa połowie częstotliwości próbkowania. Jest to najwyższa częstotliwość składowej sygnału, którą można poprawnie odwzorować przy danej częstotliwości próbkowania. Wyższe częstotliwości podlegają zjawisku aliasingu i mogą zostać błędnie odtworzone jako składowe o niższych częstotliwościach, dlatego częstotliwość odcięcia filtru musi być mniejsza od częstotliwości Nyquista.

Jeżeli liczba próbek nie przekracza długości dopełnienia wymaganej przez procedurę `filtfilt` lub częstotliwość odcięcia jest większa lub równa częstotliwości Nyquista, filtrowanie jest pomijane i zwracany jest sygnał wejściowy.

#### 4.4.2. Profile przyspieszenie–prędkość i hamowanie–prędkość

Wyznaczanie profili ASP i DSP zaimplementowano w jednej funkcji z parametrem określającym rodzaj profilu. Wpływa on na wybór minimalnej prędkości, znak przyspie-

szenia uwzględnianego w zbiorze kandydatów oraz sposób wyboru skrajnych wartości przyspieszenia w przedziałach prędkości.

Wybór kandydatów przedstawiono na listingu 6. Do dalszej analizy wybierane są próbki spełniające warunek minimalnej prędkości oraz odpowiedniego znaku przyspieszenia, dodatniego dla profilu ASP i ujemnego dla DSP.

**Listing 6.** Wybór kandydatów dla profili ASP i DSP.

```
if direction == "acc":
    min_speed = params.min_acc_speed_m_per_s
    direction_mask = accs > 0
else:
    min_speed = params.min_dec_speed_m_per_s
    direction_mask = accs < 0

candidate_indices = np.where(
    direction_mask & (speeds >= min_speed))[0]
```

Wybrane próbki dzielone są na przedziały prędkości o ustalonej szerokości. Z każdego niepustego przedziału wybierana jest określona liczba próbek o skrajnych wartościach przyspieszenia, największych dla profilu ASP i najmniejszych dla DSP. Implementację przedstawiono na listingu 7.

**Listing 7.** Podział próbek na przedziały prędkości i wybór skrajnych wartości przyspieszenia.

```
bin_edges = np.arange(
    min_speed, np.max(eligible_speeds) + bin_size, bin_size)

for lower_bound in bin_edges:
    upper_bound = lower_bound + bin_size
    bin_mask = ((eligible_speeds >= lower_bound)
                & (eligible_speeds < upper_bound))

    if not np.any(bin_mask):
        continue

    candidate_accs = eligible_accs[bin_mask]
    candidate_indices_in_bin = eligible_indices[bin_mask]
    candidate_order = np.argsort(candidate_accs)

    if direction == "acc":
        candidate_order = candidate_order[::-1]

    selected_indices.extend(
        candidate_indices_in_bin[candidate_order[:extreme_count]])
```

Do wybranych punktów dopasowywana jest regresja liniowa, a następnie identyfikowane są punkty odstające na podstawie przedziału ufności średniej odpowiedzi modelu. Implementację przedstawiono na listingu 8. W tym celu model regresji wyznaczany jest z wykorzystaniem biblioteki `statsmodels`, a dla prędkości odpowiadających analizowanemu punktowi obliczane są granice przedziału ufności. Punkty, których wartości przyspieszenia znajdują się poza tym przedziałem, są odrzucane.

**Listing 8.** Dopasowanie regresji liniowej i odrzucenie punktów spoza przedziału ufności.

```
slope , intercept = np.polyfit (
    selected_speeds , selected_accs , 1)

refit_speeds , refit_accs = _remove_confidence_interval_outliers (
    selected_speeds , selected_accs , intercept , slope ,
    confidence_level=params.confidence_level ,
)

if 2 <= len(refit_speeds) < len(selected_speeds):
    slope , intercept = np.polyfit (
        refit_speeds , refit_accs , 1)
```

Jeżeli odrzucono co najmniej jeden punkt, a w zredukowanym zbiorze pozostały co najmniej dwa punkty, regresja liniowa jest dopasowywana ponownie.

##### 4.4.3. Analiza stref boiska

Analiza przestrzenna opiera się na przekształceniu współrzędnych geograficznych do lokalnego układu współrzędnych boiska, a następnie przypisaniu każdej próbki do jednej ze stref.

Do wyznaczenia orientacji boiska z każdego pliku wybierane są próbki o prędkości nie mniejszej niż 3 m/s. Pozwala to ograniczyć wpływ fragmentów o niewielkiej intensywności ruchu na wyznaczenie orientacji lokalnego układu współrzędnych.

Jako punkt odniesienia lokalnej projekcji przyjmowana jest średnia arytmetyczna współrzędnych geograficznych wybranych próbek, oznaczona jako  $(\lambda_0, \varphi_0)$ . Współrzędne każdej próbki przeliczane są na lokalne odległości metryczne względem tego punktu zgodnie z zależnościami:

$$dx = (\lambda - \lambda_0) \cos(\varphi_0) \cdot R, \quad dy = (\varphi - \varphi_0) \cdot R,$$

gdzie  $\lambda$  i  $\varphi$  oznaczają odpowiednio długość i szerokość geograficzną wyrażone w radianach, a  $R$  jest przyjętym promieniem Ziemi.

Orientacja boiska nie jest znana z góry, dlatego kierunek jego dłuższej osi wyznaczany jest na podstawie minimalnego prostokąta ograniczającego chmurę punktów. Implementację przedstawiono na listingu 9.

**Listing 9.** Wyznaczenie kierunku głównej osi boiska metodą minimalnego prostokąta ograniczającego.

```
convex_hull_points = pitch_coordinates[
    ConvexHull(pitch_coordinates).vertices]
convex_hull_edges = np.diff(
    np.vstack([convex_hull_points, convex_hull_points[:1]]), axis=0)
edge_directions = convex_hull_edges / np.hypot(
    convex_hull_edges[:, 0], convex_hull_edges[:, 1])[:, None]

minimum_area = np.inf

for direction in edge_directions:
    perpendicular = np.array([-direction[1], direction[0]])
    along = pitch_coordinates @ direction
    across = pitch_coordinates @ perpendicular

    width = along.max() - along.min()
    height = across.max() - across.min()

    if width * height < minimum_area:
        minimum_area = width * height
        long_axis = direction if width >= height else perpendicular
```

Algorytm wykorzystuje własność minimalnego prostokąta ograniczającego, zgodnie z którą jeden z jego boków jest równoległy do jednej z krawędzi obwiedni zbioru punktów. Najpierw wyznaczana jest otoczka wypukła za pomocą funkcji `ConvexHull` z biblioteki `scipy.spatial`. Następnie dla każdej krawędzi otoczki punkty rzutowane są na osie równoległą i prostopadłą do tej krawędzi, a na podstawie zakresów obu rzutów obliczane jest pole odpowiadającego im prostokąta. Spośród analizowanych wariantów wybierany jest prostokąt o najmniejszym polu, a jego dłuższy bok wyznacza główną oś boiska.

Po wyznaczeniu orientacji lokalne współrzędne każdej próbki są obracane zgodnie z kierunkiem głównej osi boiska:

$$x = dx \cdot u_x + dy \cdot u_y + \Delta_x,$$

$$y = -dx \cdot u_y + dy \cdot u_x + \Delta_y,$$

gdzie  $\mathbf{u} = (u_x, u_y)$  oznacza kierunek głównej osi boiska, natomiast  $\Delta_x$  i  $\Delta_y$  są przesunięciami układu współrzędnych ustawiającymi środek prostokąta ograniczającego w środku boiska.

Boisko dzielone jest wzdłuż osi  $x$  na strefę lewą, środkową i prawą. Każdej próbce przypisywana jest strefa na podstawie jej współrzędnej  $x$ . Dla poszczególnych stref tworzone są maski logiczne wykorzystywane następnie do wyznaczania statystyk oraz analizy zdarzeń.

Czas przebywania zawodnika w danej strefie obliczany jest metodą trapezów zastosowaną do binarnej maski przynależności próbek:

$$T_{\text{strefa}} = \sum_{i=1}^{n-1} \frac{m_{i-1} + m_i}{2} \cdot \Delta t_i,$$

gdzie  $m_i \in \{0, 1\}$  oznacza przynależność  $i$ -tej próbki do strefy, a  $\Delta t_i$  odstęp czasowy pomiędzy kolejnymi próbkami. Jeżeli pomiędzy dwiema kolejnymi próbkami następuje zmiana strefy, metoda trapezów przypisuje analizowanej strefie połowę czasu między tymi próbkami. Odpowiada to założeniu, że przekroczenie granicy strefy następuje w połowie odstępu pomiędzy próbkami.

#### 4.4.4. Wykrywanie okien czasowych

Wykrywanie odcinków HSR oraz zdarzeń ruchowych realizowane jest za pomocą wspólnej funkcji przyjmującej warunki rozpoczęcia i zakończenia okna, minimalny czas jego trwania oraz opcjonalną maskę próbek uwzględnianych w analizie. Maską umożliwia ograniczenie wykrywania do wybranego zakresu, na przykład do próbek należących do wybranej strefy boiska.

Dla przyspieszeń i hamowań początek zdarzenia wyznacza odpowiednio dodatni lub ujemny próg przyspieszenia, a koniec przekroczenie wartości  $0 \text{ m/s}^2$ . W przypadku HSR początek odcinka wyznacza osiągnięcie progu prędkości, a koniec spadek poniżej tej wartości. Zdarzenie lub odcinek HSR jest zaliczany, gdy jego czas trwania jest nie mniejszy od ustalonej wartości minimalnej. Implementację przedstawiono na listingu 10.

**Listing 10.** Wykrywanie okien czasowych spełniających zadane warunki rozpoczęcia i zakończenia.

```
def find_windows(times, samples, start_predicate, end_predicate,
                 min_duration, mask=None):
    windows = []
    sample_count = len(samples)
    start_index = None

    for index in range(sample_count):
        sample = samples[index]
        is_sample_included = mask is None or bool(mask[index])

        if start_index is None:
            if is_sample_included and start_predicate(sample):
                start_index = index
                continue

        if not is_sample_included or end_predicate(sample):
            end_index = index - 1
            if times[end_index] - times[start_index] >= min_duration:
```

```
        windows.append((start_index, end_index))
    start_index = None

    if start_index is not None:
        end_index = sample_count - 1
        if times[end_index] - times[start_index] >= min_duration:
            windows.append((start_index, end_index))

    return windows
```

Algorytm przechodzi sekwencyjnie przez próbki szeregu czasowego. Okno rozpoczyna się od pierwszej próbki spełniającej warunek rozpoczęcia i należącej do maski uwzględnianych próbek. Kończy się na próbce poprzedzającej pierwszą próbkę spełniającą warunek zakończenia albo znajdującą się poza maską. Jeżeli czas trwania okna jest nie mniejszy od zadanego minimum, jego indeksy są zapisywane. W przypadku okna trwającego do końca szeregu czasowego jako jego koniec przyjmowana jest ostatnia próbka.

#### 4.5. Bezpieczeństwo systemu

Zakres mechanizmów bezpieczeństwa wynika z przewidywanej skali wykorzystania aplikacji oraz skoncentrowania na funkcjach analitycznych. Kontrola dostępu realizowana jest za pomocą uwierzytelniania opartego na pojedynczym hasle konfigurowanym po stronie serwera. Po podaniu poprawnego hasła serwer generuje ciasteczko sesyjne zawierające czas wystawienia oraz podpis HMAC-SHA256 obliczony z wykorzystaniem tajnego klucza przechowywanego po stronie serwera. Ciasteczko jest następnie automatycznie przesyłane przez przeglądarkę wraz z kolejnymi żądaniami, ponieważ dostęp do endpointu analizy wymaga aktywnej sesji.

Serwer nie przechowuje informacji o aktywnych sesjach. Ważność ciasteczka weryfikowana jest na podstawie jego podpisu oraz czasu wystawienia, a porównanie podpisów wykonywane jest w sposób odporny na ataki czasowe. Ograniczenie liczby prób logowania realizowane jest na poziomie serwera Nginx. Skonfigurowany limit dopuszcza do 5 żądań na minutę na adres IP z buforem 2 dodatkowych żądań, a nadmiarowe żądania są odrzucane przed przekazaniem ich do warstwy aplikacyjnej. Ogranicza to możliwość automatycznego zgadywania hasła.

Bezpieczeństwo komunikacji pomiędzy warstwą kliencką i serwerową zapewnia protokół HTTPS wykorzystujący TLS. W środowisku produkcyjnym ciasteczko sesyjne oznaczone jest atrybutami `HttpOnly`, `Secure` oraz `SameSite=None`. Atrybut `HttpOnly` uniemożliwia dostęp do ciasteczka z poziomu JavaScript, ograniczając możliwość jego odczytania w przypadku ataku XSS. Atrybut `Secure` powoduje przesyłanie ciasteczka wyłącznie za pośrednictwem HTTPS, natomiast `SameSite=None` umożliwia jego przesyłanie w komunikacji pomiędzy osobno wdrożonymi warstwami systemu. Dostęp do interfejsu API ograniczony jest dodatkowo przez konfigurację CORS, dopuszczającą żądania pochodzące z określonych domen aplikacji klienckiej.

#### 4. Projekt i implementacja systemu

---

Dane są przetwarzane w ramach żądania, a otrzymane wyniki zwracane bezpośrednio do aplikacji klienckiej. Brak trwałego przechowywania ogranicza ryzyko nieuprawnionego dostępu do danych zawodników.

## 5. Weryfikacja rozwiązania

Pierwsza wersja systemu umożliwiała filtrowanie prędkości, wyznaczanie profilu ASP oraz interaktywną prezentację wyników w postaci tabeli statystyk, szeregów czasowych, trajektorii zawodnika i histogramów. Na podstawie uwag eksperta z klubu piłkarskiego dodano filtr Butterwortha, podział boiska na strefy, wykrywanie odcinków HSR, wyznaczenie profilu DSP oraz analizę zdarzeń przyspieszenia i hamowania z uwzględnieniem fazy wczesnej i późnej, przedziałów intensywności oraz parametrów mechanicznych.

### 5.1. Testy jednostkowe i integracyjne warstwy serwerowej

Testy warstwy serwerowej przeprowadzono z wykorzystaniem biblioteki `pytest` oraz narzędzia `coverage.py` do pomiaru pokrycia kodu.

W testach integracyjnych interfejsu REST API użyto testowego klienta FastAPI. Zweryfikowano obsługę żądań analizy, walidację danych wejściowych, obsługę błędów pojedynczych plików oraz nadpisywanie parametrów. Sprawdzono logowanie, ustawienie ciasteczka sesyjnego oraz weryfikację sesji.

Wyniki pomiaru pokrycia kodu warstwy serwerowej przedstawiono w tabeli 5.1.

**Tabela 5.1.** Pokrycie kodu warstwy serwerowej testami.

| Rodzaj pokrycia | Wartość | Pokryte / całość |
|-----------------|---------|------------------|
| Pokrycie linii  | 100,00% | 610 / 610        |
| Pokrycie gałęzi | 100,00% | 122 / 122        |

Testy pokryły wszystkie linie i gałęzie kodu.

### 5.2. Porównanie z referencyjną implementacją profilu ASP

Jako punkt odniesienia wykorzystano pakiet `InSituASProfile` [9]. Porównanie przeprowadzono na przykładowych danych z jego repozytorium, ustawiając w `ASProfiler` parametry zgodnie z wartościami stosowanymi w implementacji referencyjnej, czyli minimalną prędkość 3 m/s, szerokość przedziału 0,1 m/s, dwie najwyższe wartości przyspieszenia w każdym przedziale oraz poziom ufności 95%.

Wyniki porównania z implementacją referencyjną przedstawiono w tabeli 5.2.

**Tabela 5.2.** Porównanie wyników profilu ASP z implementacją `InSituASProfile`.

| Metryka                                  | <code>InSituASProfile</code> | <code>ASProfiler</code> |
|------------------------------------------|------------------------------|-------------------------|
| $A_0$ [m/s <sup>2</sup> ]                | 8,273012                     | 8,273012                |
| $V_0$ [m/s]                              | 8,444761                     | 8,444761                |
| $R^2$                                    | 0,998607                     | 0,998607                |
| Liczba punktów po odrzuceniu odstających | 29                           | 29                      |

Obie implementacje zachowały tyle samo punktów po odrzuceniu obserwacji odstających i uzyskały zgodne parametry profilu.

### 5.3. Testy warstwy klienckiej

Warstwę kliencką przetestowano z wykorzystaniem środowiska Vitest, mechanizmu V8 do pomiaru pokrycia kodu oraz biblioteki React Testing Library do testowania komponentów. Testy jednostkowe obejmują komunikację z interfejsem API, zarządzanie stanem, przygotowanie danych do wizualizacji, eksport wyników oraz wybrane komponenty interfejsu użytkownika.

W testach integracyjnych komponentów użyto testowej implementacji komunikacji z interfejsem API. Zweryfikowano współpracę widoków z mechanizmami zarządzania stanem, routingiem oraz komunikacją z interfejsem API.

Pomiar pokrycia kodu przeprowadzono dla katalogów `src/api/`, `src/utils/` oraz `src/hooks/`, zawierających główną logikę warstwy klienckiej.

Wyniki pomiaru pokrycia kodu warstwy klienckiej przedstawiono w tabeli 5.3.

**Tabela 5.3.** Pokrycie kodu warstwy klienckiej w katalogach `api/`, `hooks/` i `utils/`.

| Rodzaj pokrycia  | Wartość | Pokryte / całość |
|------------------|---------|------------------|
| Pokrycie linii   | 100,00% | 865 / 865        |
| Pokrycie funkcji | 100,00% | 403 / 403        |
| Pokrycie gałęzi  | 95,99%  | 407 / 424        |

Uzyskano 100% pokrycia linii i funkcji oraz 95,99% pokrycia gałęzi.

### 5.4. Weryfikacja end-to-end

Weryfikację end-to-end przeprowadzono ręcznie na rzeczywistych plikach z platformy STATSports, przekazanych przez klub sportowy. Sprawdzono logowanie, analizę pojedynczego i wielu plików, konfigurację parametrów wspólnych i indywidualnych, filtrowanie na podstawie jakości sygnału GNSS, wyznaczanie profili ASP i DSP, wykrywanie zdarzeń przyspieszenia i hamowania oraz ich analizę w kontekście stref boiska i odcinków HSR.

Zweryfikowano również prezentację wyników, powiązanie zaznaczonych próbek pomiędzy wykresami, eksport danych oraz obsługę błędów pojedynczych plików bez przerywania analizy pozostałych.

## 6. Podsumowanie

Zaprojektowano i zaimplementowano aplikację umożliwiającą analizę danych GNSS zbieranych od piłkarzy. System wspiera ocenę intensywności aktywności zawodników i porównywanie wyników uzyskiwanych dla różnych parametrów analizy.

Spełniono wymagania dotyczące dostępności aplikacji, przechowywania danych, uwierzytelniania, szyfrowania komunikacji, czasu przetwarzania danych, dostępności kodu źródłowego, rozszerzalności oraz niskich kosztów. System umożliwia uwierzytelnienie, konfigurację parametrów, wczytanie i przygotowanie danych, analizę przestrzenną i HSR, analizę przyspieszeń i hamowań, prezentację oraz eksport wyników.

Kod źródłowy aplikacji udostępniono w publicznym repozytorium GitHub: <https://github.com/sz-aleksandra/asprofiler>.

Poprawność implementacji zweryfikowano testami jednostkowymi i integracyjnymi warstwy serwerowej, porównaniem z referencyjną implementacją profilu ASP, testami warstwy klienckiej oraz ręczną weryfikacją end-to-end na rzeczywistych danych.

Rozbudowana konfiguracja parametrów analizy pozwala badać wpływ ustawień, które w systemach komercyjnych często pozostają niedostępne dla użytkownika. Dotyczy to między innymi minimalnego czasu trwania zdarzeń oraz parametrów wyznaczania profilu ASP i FVP. Zastosowanie tego podejścia do profilowania hamowań umożliwia ocenę, czy zaproponowany profil DSP może wspierać opis mechaniki hamowania. Analiza faz hamowań pozwala weryfikować teorię dotyczącą ich przebiegu w praktyce.

Głównymi ograniczeniami aplikacji są konieczność ręcznego wgrywania plików przed każdą analizą oraz uproszczony mechanizm uwierzytelniania. Wynika to z przyjętego zakresu pracy i skupienia na warstwie analitycznej. System obsługuje wyłącznie pliki CSV eksportowane z platformy STATSports. Nie umożliwia pracy z wybranymi fragmentami aktywności i wieloma sesjami oraz konfiguracji progów przedziałów intensywności. Modułowa organizacja systemu ułatwia jego dalsze rozszerzanie.

Głównym kierunkiem dalszego rozwoju, wskazanym przez eksperta z klubu sportowego, jest integracja z API STATSports lub usługami przechowywania plików w chmurze.



## Bibliografia

- [1] Catapult Sports, *Vector OpenField*, Dostęp zdalny (08.06.2026): <https://core.catapultsports.com/hc/en-us/categories/7209563887631-Vector-OpenField-Software>, 2024.
- [2] Catapult Sports, *Athlete Monitoring System*, Dostęp zdalny (06.09.2026): <https://www.catapult.com/solutions/athlete-monitoring>, 2026.
- [3] P. Cormier i in., „Concurrent validity and reliability of different technologies for sprint-derived horizontal force-velocity-power profiling”, *Journal of Strength and Conditioning Research*, t. 37, nr 6, s. 1298–1305, 2023. DOI: 10.1519/JSC.00000000000004429
- [4] Exelio SRL, *gpexe UNO System*, Dostęp zdalny (09.06.2025): <https://www.gpexe.com/uno-system/>, 2025.
- [5] A. Ferraz, P. Duarte-Mendes, H. Sarmento, J. Valente-Dos-Santos i B. Travassos, „Tracking devices and physical performance analysis in team sports: a comprehensive framework for research—trends and future directions”, *Frontiers in Sports and Active Living*, t. 5, s. 1284086, 2023. DOI: 10.3389/fspor.2023.1284086
- [6] A. Gualtieri, E. Rampinini, A. Dello Iacono i M. Beato, „High-speed running and sprinting in professional adult soccer: Current thresholds definition, match demands and training strategies. A systematic review”, *Frontiers in Sports and Active Living*, t. 5, s. 1116293, 2023. DOI: 10.3389/fspor.2023.1116293
- [7] D. J. Harper, C. Carling i J. Kiely, „High-intensity acceleration and deceleration demands in elite team sports competitive match play: a systematic review and meta-analysis of observational studies”, *Sports Medicine*, t. 49, s. 1923–1947, 2019. DOI: 10.1007/s40279-019-01170-1
- [8] D. J. Harper, J.-B. Morin, C. Carling i J. Kiely, „Measuring maximal horizontal deceleration ability using radar technology: reliability and sensitivity of kinematic and kinetic variables”, *Sports Biomechanics*, t. 22, nr 9, 2020. DOI: 10.1080/14763141.2020.1792968
- [9] L. Ishøi, *InSituASProfile*, Dostęp zdalny (10.09.2026): <https://github.com/LasseIshoi/InSituASProfile>, 2022.
- [10] J. Lutz, D. Memmert, D. Raabe, R. Dornberger i L. Donath, „Wearables for integrative performance and tactic analyses: opportunities, challenges, and future directions”, *International Journal of Environmental Research and Public Health*, t. 17, nr 1, s. 59, 2019. DOI: 10.3390/ijerph17010059
- [11] J. J. Malone, R. Lovell, M. C. Varley i A. J. Coutts, „Unpacking the black box: applications and considerations for using GPS devices in sport”, *International Journal of Sports Physiology and Performance*, t. 12, nr Suppl 2, S2-18–S2-26, 2017. DOI: 10.1123/ijssp.2016-0236
- [12] J.-B. Morin i in., „Individual acceleration-speed profile in-situ: a proof of concept in professional football players”, *Journal of Biomechanics*, t. 123, s. 110524, 2021. DOI: 10.1016/j.jbiomech.2021.110524

- [13] S. Robertson i in., „Challenges and considerations in determining the quality of electronic performance & tracking systems for team sports”, *Frontiers in Sports and Active Living*, t. 5, s. 1 266 522, 2023. DOI: 10.3389/fspor.2023.1266522
- [14] M. T. U. Scott, T. J. Scott i V. G. Kelly, „The validity and reliability of global positioning systems in team sport: a brief review”, *Journal of Strength and Conditioning Research*, t. 30, nr 5, s. 1470–1490, 2016. DOI: 10.1519/JSC.0000000000001221
- [15] STATSports, *STATSports Launch The Most Powerful Wearable In Sport*, Dostęp zdalny (06.09.2026): <https://statsports.com/article/statsports-launch-the-most-powerful-wearable-in-sport>, 2020.
- [16] STATSports, *AI-Powered Insights Added to Sonra Lite: Smarter, Faster, Easier Analysis for Grassroots Coaches*, Dostęp zdalny (08.09.2026): <https://statsports.com/article/ai-powered-insights-added-to-sonra-lite-smarter-faster-easier-analysis-for-grassroots-coaches>, 2025.
- [17] STATSports, *Introducing Force Velocity Profiling: Unlocking Sprint Performance Insights*, Dostęp zdalny (31.08.2026): <https://statsports.com/article/introducing-force-velocity-profiling-unlocking-sprint-performance-insights>, 2025.
- [18] STATSports, *STATSports Elite Support Centre*, Dostęp zdalny (08.06.2026): <https://elitesupport.statsports.com/hc/en-us>, 2026.
- [19] L. Torres-Ronda, E. Beanland, S. Whitehead, A. Sweeting i J. Clubb, „Tracking Systems in Team Sports: A Narrative Review of Applications of the Data and Sport Specific Analysis”, *Sports Medicine - Open*, t. 8, nr 15, 2022. DOI: 10.1186/s40798-022-00408-z
- [20] M. C. Varley, A. Jaspers, W. F. Helsen i J. J. Malone, „Methodological considerations when quantifying high-intensity efforts in team sport using global positioning system technology”, *International Journal of Sports Physiology and Performance*, t. 12, nr 8, s. 1059–1068, 2017. DOI: 10.1123/ijsp.2016-0534
- [21] T. P. Zieliński, *Cyfrowe przetwarzanie sygnałów: Od teorii do zastosowań*. Warszawa: Wydawnictwa Komunikacji i Łączności, 2005.

## Wykaz symboli i skrótów

|              |                                                             |
|--------------|-------------------------------------------------------------|
| <b>AI</b>    | – ang. <i>Artificial Intelligence</i>                       |
| <b>API</b>   | – ang. <i>Application Programming Interface</i>             |
| <b>ASGI</b>  | – ang. <i>Asynchronous Server Gateway Interface</i>         |
| <b>ASP</b>   | – ang. <i>Acceleration–Speed Profile</i>                    |
| <b>AWS</b>   | – ang. <i>Amazon Web Services</i>                           |
| <b>CORS</b>  | – ang. <i>Cross-Origin Resource Sharing</i>                 |
| <b>CSV</b>   | – ang. <i>Comma-Separated Values</i>                        |
| <b>DSP</b>   | – ang. <i>Deceleration–Speed Profile</i>                    |
| <b>EC2</b>   | – ang. <i>Elastic Compute Cloud</i>                         |
| <b>FVP</b>   | – ang. <i>Force–Velocity Profile</i>                        |
| <b>GNSS</b>  | – ang. <i>Global Navigation Satellite System</i>            |
| <b>GPS</b>   | – ang. <i>Global Positioning System</i>                     |
| <b>Hacc</b>  | – ang. <i>Horizontal Accuracy</i>                           |
| <b>HDOP</b>  | – ang. <i>Horizontal Dilution of Precision</i>              |
| <b>HSR</b>   | – ang. <i>High-Speed Running</i>                            |
| <b>HTTPS</b> | – ang. <i>HyperText Transfer Protocol Secure</i>            |
| <b>IIR</b>   | – ang. <i>Infinite Impulse Response</i>                     |
| <b>IMU</b>   | – ang. <i>Inertial Measurement Unit</i>                     |
| <b>JSON</b>  | – ang. <i>JavaScript Object Notation</i>                    |
| <b>ML</b>    | – ang. <i>Machine Learning</i>                              |
| <b>REST</b>  | – ang. <i>Representational State Transfer</i>               |
| <b>S3</b>    | – ang. <i>Simple Storage Service</i>                        |
| <b>TLS</b>   | – ang. <i>Transport Layer Security</i>                      |
| <b>V8</b>    | – silnik JavaScript wykorzystywany do pomiaru pokrycia kodu |
| <b>XSS</b>   | – ang. <i>Cross-Site Scripting</i>                          |
| <b>ZIP</b>   | – format archiwum plików                                    |

## Spis rysunków

|     |                                                                                                           |    |
|-----|-----------------------------------------------------------------------------------------------------------|----|
| 2.1 | Porównanie średniej ruchomej, filtru medianowego i filtru Butterwortha na sygnale prędkości GNSS. . . . . | 10 |
| 2.2 | Profil ASP wraz z etapami jego wyznaczania. . . . .                                                       | 11 |
| 4.1 | Diagram C4 kontenerów projektowanego systemu. . . . .                                                     | 18 |
| 4.2 | Diagram C3 komponentów warstwy klienckiej. . . . .                                                        | 19 |
| 4.3 | Widok logowania. . . . .                                                                                  | 20 |
| 4.4 | Widok informacyjny. . . . .                                                                               | 20 |
| 4.5 | Widok listy plików. Dodawanie plików i formularz konfiguracji globalnych parametrów analizy. . . . .      | 20 |

|      |                                                                                                                                               |    |
|------|-----------------------------------------------------------------------------------------------------------------------------------------------|----|
| 4.6  | Widok listy plików. Wybór i nadpisywanie parametrów analizy dla poszczególnych plików. . . . .                                                | 21 |
| 4.7  | Widok analizy. Parametry profili i ustawienia widoku. . . . .                                                                                 | 21 |
| 4.8  | Widok analizy. Wykres profilu ASP z próbkami zaznaczonymi przez użytkownika i wyróżnionymi punktami sąsiednimi. . . . .                       | 22 |
| 4.9  | Widok analizy. Konfiguracja liczby punktów wyróżnianych wokół zaznaczonej próbki i zarządzanie zaznaczonymi próbkami w pasku bocznym. . . . . | 22 |
| 4.10 | Widok analizy. Tabela ogólnych statystyk. . . . .                                                                                             | 23 |
| 4.11 | Widok analizy. Szereg czasowy prędkości z wyróżnionymi punktami. . . . .                                                                      | 23 |
| 4.12 | Widok analizy. Trajektoria zawodnika na tle boiska z wyróżnionymi punktami. . . . .                                                           | 24 |
| 4.13 | Widok analizy. Histogram zdarzeń hamowania. . . . .                                                                                           | 24 |
| 4.14 | Widok analizy. Statystyki wykrytych zdarzeń przyspieszenia z podziałem na fazy. . . . .                                                       | 24 |
| 4.15 | Diagram C3 komponentów warstwy serwerowej. . . . .                                                                                            | 25 |
| 4.16 | Diagram modułów usługi analizy. . . . .                                                                                                       | 26 |

## Spis tabel

|     |                                                                              |    |
|-----|------------------------------------------------------------------------------|----|
| 4.1 | Parametry konfiguracji wykorzystywane przez system. . . . .                  | 28 |
| 5.1 | Pokrycie kodu warstwy serwerowej testami. . . . .                            | 39 |
| 5.2 | Porównanie wyników profilu ASP z implementacją InSituASProfile. . . . .      | 39 |
| 5.3 | Pokrycie kodu warstwy klienckiej w katalogach api/, hooks/ i utils/. . . . . | 40 |