

**Dokumentacja dla projektu „Runnin”
w ramach zajęć Inżynierii oprogramowania
w semestrze zimowym
roku akademickiego 2023/2024**

**Filip Kokot
Michał Tyszecki
Łukasz Nosarzewski**

SPIS TREŚCI

1. Charakterystyka oprogramowania	3
1.1. Nazwa skrócona	3
1.2. Nazwa pełna	3
1.3. Krótki opis ze wskazaniem celów	3
1.4. Link do repozytorium	3
2. Prawa autorskie	3
2.1. Autorzy	3
2.2. Warunki licencyjne oprogramowania	3
3. Specyfikacja wymagań	4
4. Architektura oprogramowania	6
4.1. Architektura rozwoju – stos technologiczny	6
4.2. Architektura uruchomieniowa – stos technologiczny	6
5. Testy	7
5.1. Scenariusze testów	7
5.2. Sprawozdanie z wykonania scenariuszy testów	8
Źródła	10

1. CHARAKTERYSTYKA OPROGRAMOWANIA

1.1. Nazwa skrócona

„Runnin”

1.2. Nazwa pełna

System monitorowania treningu personalnego z aplikacją Strava

1.3. Krótki opis ze wskazaniem celów

Projekt "Runnin" został stworzony jako zaawansowane narzędzie wspierające trenerów personalnych w monitorowaniu i analizie danych treningowych ich podopiecznych. Głównym celem systemu jest integracja z platformą Strava, aby umożliwić automatyczne zbieranie i przetwarzanie aktywności sportowych użytkowników. Aplikacja ma na celu umożliwienie trenerom dostępu do szczegółowych danych o treningach, postępach i osiągnięciach swoich klientów w sposób łatwy, szybki i efektywny.

1.4. Link do repozytorium

<https://github.com/Michaltys/Runnin>

2. PRAWA AUTORSKIE

2.1. Autorzy

- Michał Tyszecki - programista
- Filip Kokot – analityk danych
- Łukasz Nosarzewski – analityk biznesowy

2.2. Warunki licencyjne do oprogramowania

MIT License → licencja otwartego oprogramowania dająca użytkownikom nieograniczone prawo do używania, kopiowania, modyfikowania i rozpowszechniania oryginalnego lub zmodyfikowanego programu.

3. SPECYFIKACJA WYMAGAŃ

Wymagania funkcjonalne:

1 – Integracja z aplikacją Strava: autoryzacja podopiecznego, pobranie danych z API Strava dla danego użytkownika, zrzut danych do pliku,

2 – Wyświetlenie raportu w Power BI

Identyfikator	WF1
Nazwa	Integracja z aplikacją Strava
Opis	Integracja z platformą Strava umożliwia trenerom personalnym dostęp do danych treningowych ich podopiecznych. Stworzenie środowiska do łączenia z API, postawienie aplikacji webowej. Backend aplikacji umożliwia trenerom zarządzanie listą podopiecznych, w tym dodawanie, edycję i usuwanie profili.
Priorytet	1
Kategoria	Funkcjonalne

Identyfikator	WF1.1
Nazwa	Autoryzacja podopiecznego
Opis	Umożliwia podopiecznym autoryzację ich kont Strava w aplikacji, co pozwala trenerowi na dostęp do ich danych treningowych. Autoryzacja odbywa się poprzez bezpieczny proces OAuth 2.0.
Priorytet	1
Kategoria	Funkcjonalne

Identyfikator	WF1.2
Nazwa	Pobranie danych z API Strava dla danego użytkownika
Opis	Funkcjonalność pozwala na pobieranie i przetwarzanie danych z API Strava, w tym aktywności treningowych, tras i statystyk, co jest kluczowe dla analizy wydajności i postępów podopiecznych.
Priorytet	1
Kategoria	Funkcjonalne

Identyfikator	WF1.3
Nazwa	Zrzut danych do pliku, stworzenie bazy i połączenie jej z PowerBI
Opis	Dane powinny być zrzucane do pliku z rozszerzeniem .sql, który po odpaleniu automatycznie tworzy nową bazę. W dalszej kolejności następuje połączenie bazy z PowerBI, gdzie są eksportowane aktywności.
Priorytet	1
Kategoria	Funkcjonalne

Identyfikator	WF2
Nazwa	Wyświetlenie raportu w PowerBI
Opis	Trenerzy mają dostęp do dashboardów w PowerBI, które wizualizują dane treningowe podopiecznych, umożliwiając łatwą analizę ich postępów i dostosowanie planów treningowych.
Priorytet	1
Kategoria	Funkcjonalne

Wymagania niefunkcjonalne:

1 – Skalowalność, intuicyjność i łatwość obsługi

2 – Dostępność na różnych urządzeniach

Identyfikator	WNF1
Nazwa	Skalowalność
Opis	System powinien być zaprojektowany z myślą o skalowalności, aby mógł obsłużyć rosnącą liczbę podopiecznych i trenerów bez utraty wydajności. Powinien być elastyczny w zakresie przetwarzania dużej ilości danych treningowych i zapewniać szybką odpowiedź.
Priorytet	1
Kategoria	Niefunkcjonalne

Identyfikator	WNF1.1
Nazwa	Intuicyjność i łatwość obsługi
Opis	Interfejs użytkownika powinien być intuicyjny i łatwy w obsłudze, umożliwiając trenerom szybki dostęp do potrzebnych funkcji i danych bez konieczności szczegółowego szkolenia.
Priorytet	1
Kategoria	Niefunkcjonalne

Identyfikator	WNF2
Nazwa	Dostępność na różnych urządzeniach
Opis	Aplikacja powinna być dostępna i w pełni funkcjonalna na różnych urządzeniach, w tym na komputerach stacjonarnych, laptopach, tabletach i smartfonach. Interfejs użytkownika powinien być responsywny i dostosowany do różnych rozdzielczości ekranu.
Priorytet	2
Kategoria	Niefunkcjonalne

4. ARCHITEKTURA OPROGRAMOWANIA

4.1. Architektura rozwoju – stos technologiczny

- Python (Django)
- Baza danych: MySQL
- HTML + CSS
- System kontroli wersji: GitHub
- MS PowerBI

4.2. Architektura uruchomieniowa – stos technologiczny

- Windows
- PythonAnywhere
- Bash

5. TESTY

5.1. Scenariusze testów

Identyfikator	TWF1 i TWF1.1
Nazwa scenariusza	Dodawanie użytkownika (autoryzacja)
Scenariusz	Scenariusz dotyczy sprawdzenia, jak działa integracja z aplikacją Strava, kiedy trener próbuje dodać nowego podopiecznego. Kluczowym elementem jest tu proces autoryzacji, podczas którego trener podejmuje kroki, by dodać nowego użytkownika i zintegrować jego konto z aplikacją Strava.

Identyfikator	TWF1.2
Nazwa scenariusza	Pobranie danych z API Strava dla danego użytkownika podanego w TWF1 i TWF1.1
Scenariusz	Scenariusz obejmuje weryfikację pojawienia się nowego użytkownika w aplikacji stworzonej dzięki Python Anywhere.

Identyfikator	TWF2
Nazwa scenariusza	Odświeżenie raportu w PowerBI
Scenariusz	Scenariusz obejmuje odświeżenie raportu w PowerBI i sprawdzenie czy dane o nowym użytkowniku zaktualizowały się poprawnie.

Identyfikator	TWF3
Nazwa scenariusza	Test wizualizacji w PowerBI
Scenariusz	Ogólna weryfikacja raportu, gdzie tester powinien sprawdzić wszystkie wizualizacje pod kątem poprawności merytorycznej i graficznej.

Identyfikator	TWNF1
Nazwa scenariusza	Testowanie skalowalności
Scenariusz	Skalowalność powinna zostać przetestowana w ramach scenariusza TWF1 i TWF1.1

Identyfikator	TWNF1.1
Nazwa scenariusza	Testowanie intuicyjności i łatwości obsługi
Scenariusz	Przeprowadzenie testu powinno obejmować udostępnienie raportu osobie, która nigdy wcześniej z niego nie korzystała. Celem jest, aby ta osoba, będąca testerem, oceniła, jak łatwo jest obsługiwać ten raport i czy jego użycie jest intuicyjne, szczególnie z punktu widzenia kogoś, kto nie ma zaawansowanej wiedzy na temat używanego oprogramowania.

Identyfikator	TWNF2
Nazwa scenariusza	Testowanie dostępności na różnych urządzeniach
Scenariusz	Test ma za zadanie zweryfikować, czy aplikacja działa poprawnie na urządzeniach mobilnych i stacjonarnych.

5.2. Sprawozdanie z wykonania scenariuszy testów

Identyfikator	TWF1 i TWF1.1
Nazwa scenariusza	Dodawanie użytkownika (autentykacja)
Sprawozdanie	Tester dodał użytkownika „Filip Kokot” do systemu za pomocą autoryzacji Strava. Autoryzacja przebiegła pomyślnie. Test przebiegł pomyślnie.

Identyfikator	TWF1.2
Nazwa scenariusza	Pobranie danych z API Strava dla danego użytkownika podanego w TWF1 i TWF1.1
Sprawozdanie	Tester sprawdził, iż „Filip Kokot” został dodany do listy sportowców. Test przebiegł pomyślnie.

Identyfikator	TWF2
Nazwa scenariusza	Odświeżenie raportu w PowerBI
Sprawozdanie	<ol style="list-style-type: none"> 1. Tester włączył raport „Runnin” w PowerBI, a następnie odświeżył dane w raporcie klikając przycisk „Refresh” 2. Dane nt. użytkownika „Filip Kokot” zostały dodane. Test przebiegł pomyślnie.

Identyfikator	TWNF1
Nazwa scenariusza	Testowanie skalowalności
Sprawozdanie	Testy skalowalności zostały przeprowadzone w poprzednich scenariuszach testowych, poprzez dodanie podopiecznego. Dodanie danych personalnych przebiegło pomyślnie.

Identyfikator	TWNF1.1
Nazwa scenariusza	Testowanie intuicyjności i łatwości obsługi
Sprawozdanie	<ol style="list-style-type: none"> 1. Osoba testująca aplikację autoryzowała swoje konto. Łatwość obsługi oceniła na „bardzo proste” 2. Osoba testująca włączyła listę sportowców. Łatwość obsługi i intuicyjność oceniła na „bardzo proste” 3. Osoba testująca spróbowała przenieść dane do PowerBI. Niestety, wywoływanie komend w Bash oraz zrzucanie bazy danych do MySQL nie udało się. Autorzy aplikacji musieli zrobić to za osobę testującą. Osoba testująca oceniła łatwość obsługi na „bardzo słabe” 4. Tester odświeżył raport w PowerBI i znalazł dodane przez siebie dane. Ocena: „bardzo proste” <p>Test nie przeszedł pomyślnie. Zrzucanie danych do PowerBI okazało się zbyt skomplikowane. Docelowo, dane te będą zrzucane się automatycznie, więc rezultat ten był zgodny z przewidywaniami.</p>

Identyfikator	TWNF2
Nazwa scenariusza	Testowanie dostępności na różnych urządzeniach
Sprawozdanie	<p>Test wykazał, że aplikacja częściowo działa na różnych urządzeniach. O ile część zależna od autorów, czyt. Autoryzacja ze strawą i dodawanie użytkownika, zrzucanie danych działają na różnych urządzeniach, o tyle raport w PowerBI działa tylko na komputerze.</p> <p>Test nie przeszedł pomyślnie. Jest to spowodowane cennikiem PowerBI. Korzystanie z raportów w tym środowisku na telefonie dostępne jest tylko w przypadku typu konta „premium”, które jest dodatkowo płatne. Oczywiście, jest to możliwe do poprawienia w przyszłości, poprzez zakup poprawnej licencji PowerBI.</p>

ŹRÓDŁA

1. Dokumentacja aplikacji Strava:

- https://developers.strava.com/docs/authentication/?fbclid=IwAR0pYyPruykMYZQuXplOGRYmmLBa5CfeMW9kN6Yt9x8RKzkOphCKXdIa7_A
- <https://developers.strava.com/docs/reference/?fbclid=IwAR3f8IbjxwYI7Ook5GwnfSBC99IU6Sy3YJje0apf1wwiT3xO7raDnlzFHI#api-Activities-getActivityById>
- <https://www.strava.com/features>