|  |  |
| --- | --- |
| **Politechnika Świętokrzyska w Kielcach Wydział**  **Elektrotechniki, Automatyki i Informatyki** | |
| **Aplikacje Mobilne - Projekt**  Rok akademicki - 2024/2025 | |
| **Temat projektu:**  Aplikacja do Treningu | **Wykonali:**  Stępień Filip  Karwat Damian  Grot Rafał  **Grupa:** 3ID11B |

**SPIS TREŚCI**

|  |  |
| --- | --- |
| 1. Wstęp | 3 |
| * 1. Krótki opis aplikacji | 3 |
| * 1. Wykorzystana technologia i narzędzia | 4 |
| * 1. Wykorzystane natywne funkcje urządzenia | 5 |
| 1. Implementacja | 6 |
| * 1. Ekrany dostępne w aplikacji | 6 |
| * 1. Link do repozytorium | 14 |
| 1. Testy aplikacji | 15 |
| 1. Podział pracy | 18 |
| 1. Podsumowanie | 18 |

1. **Wstęp**
   1. **Krótki opis aplikacji**

Zadaniem stworzonej aplikacji jest asystowanie w śledzeniu postępów, oraz możliwość prowadzenia dziennika dla treningów na siłowni. Aplikacja udostępnia poniżej wymienione funkcjonalności:

* Logowanie i Rejestracja,
  + Użytkownik może stworzyć własne konto, lub zalogować się na istniejące.
  + Do każdego konta przypisane są dane użytkownika, takie jak statystyki, poprzednio wykonane treningi, oraz dane logowania.
  + Aplikacja posiada funkcje szybkiego logowania się – logowanie następuje automatyczne, jeśli na danym urządzeniu ktoś był zalogowany wcześniej.
* Edycja danych logowania,
  + Użytkownik może zmienić swój login i hasło.
* Śledzenie statystyk dotyczących ćwiczeń i treningów,
  + Na profilu użytkownika dostępne są statystyki z wykonanych treningów,
  + Na ekranie głównym znajduje się lista poprzednich treningów, zawierająca ich szczegóły.
* Przeglądanie predefiniowanych ćwiczeń,
  + Na ekranie ćwiczeń znajduję się lista ćwiczeń dostępnych   
    w aplikacji, każde ćwiczenie ma przypisane mięśnie na które oddziałuje, oraz instrukcje które można przeczytać po kliknięciu.
* Tworzenie treningu z gotowych ćwiczeń,
  + Podczas treningu użytkownik może dodawać do niego dowolną ilość ćwiczeń z biblioteki ćwiczeń,
  + Dla każdego wykonywanego ćwiczenia użytkownik dodaje ilość serii, powtórzeń, oraz wagę jakiej użył.
* Zapisywanie treningów w kalendarzu.
  + Treningi zapisywane są w kalendarzu ćwiczeń, umożliwiając użytkownikowi późniejsze podejrzenie w jakich dniach trenował, oraz ile dni odpoczywał, bądź ile dni z rzędu odbył trening.
  1. **Wykorzystana technologia i narzędzia**

Do stworzenia aplikacji wykorzystano następujące technologie i narzędzia:

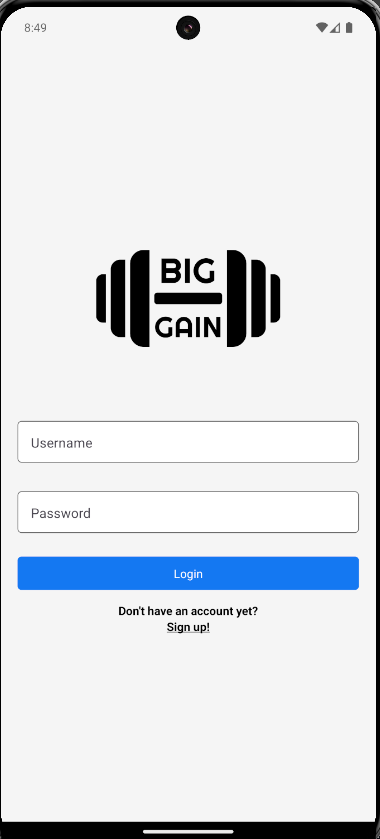
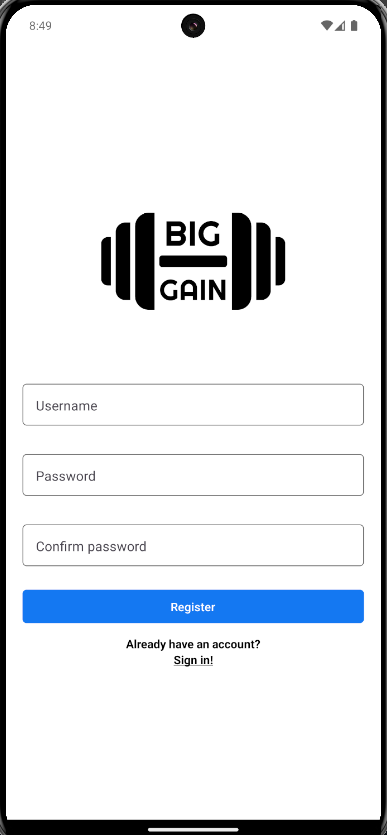
* **Środowisko Expo** w wersji ~52.0.4 - umożliwia łatwą konfigurację   
  i uruchamianie aplikacji mobilnych w oparciu o React Native, oraz udostępnia wiele dodatkowych narzędzi i interfejsów, takich jak expo-secure-store.
* **Język programowania TypeScript** w wersji ^5.3.3, oraz jego biblioteka React w wersji 18.3.1,. Do tworzenia aplikacji mobilnej wykorzystano **React Native** w wersji 0.76.1,
* **React Navigation** w wersjach 7.0.0, (Stack Navigation) oraz 7.0.1 (Bottom Tab Navigation), który odpowiada za zarządzanie nawigacją między ekranami aplikacji.
* **Biblioteka React Native Reanimated** w wersji ~3.16.1, używana do tworzenia płynnych animacji w interfejsie użytkownika. W aplikacji użyta do ustawienia motywów, oraz do tworzenia tabel.
* **Expo Image Picker** w wersji ~16.0.3, pozwalający na wybieranie obrazów z galerii urządzenia lub robienie zdjęć aparatem. W aplikacji użyta do dodawania zdjęć przypisanych do danych treningów.
* **Expo Secure Store** w wersji ~14.0.0, wykorzystywany do bezpiecznego przechowywania danych użytkownika.
* **Axios** w wersji ^1.7.8, który służy do obsługi zapytań HTTP i komunikacji z API.
* **Day.js** w wersji ^1.11.13, lekka biblioteka do manipulacji i formatowania dat przypisanych do treningów,
* **Victory Native** w wersji ^41.14.0, jest to biblioteka służąca do tworzenia wykresów, w tym przypadku wykresu kołowego i liniowego,
* **React Native Skia** w wersji ^1.7.3, biblioteka służąca do generowania grafik 2d w React-Native, używana przy tworzeniu wykresu rozkładu treningów
* **Fuse.js** w wersji ^7.0.0, do implementacji wyszukiwania w czasie rzeczywistym, użyta do inteligentnego wyszukiwana ćwiczeń,
* **React Native Paper** w wersji ^5.12.5, która dostarcza gotowe komponenty UI zgodne z Material Design, zostały użyte z niej komponenty jak   
  i motywy,
* **Json-server** w wersji ^1.0.0-beta.3, używany do tworzenia lokalnego serwera API w celach testowych.
* **Narzędzia ESLint** (^8.57.0) i **Prettier** (^3.3.3), które zapewniają jednolitą jakość i formatowanie kodu.
* **Babel** w wersji ^7.25.2, służący do transpilacji nowoczesnego JavaScriptu na kod kompatybilny z urządzeniami mobilnymi.
  1. **Wykorzystane natywne funkcje urządzenia**

W aplikacji wykorzystano następujące funkcje natywne urządzenia:

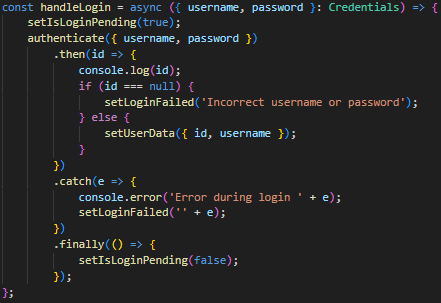
* Aparat - **Expo Image Picker –** umożliwia użytkownikowi aplikacji dodawanie zdjęć z treningu, i zapisywanie go.
* Wibracje – **React Native Vibration –** wibracje urządzenia uruchamiają się, gdy dana część treningu się skończy, w formie przypomnienia, że należy kontynuować trening.

1. **Implementacja**
   1. **Ekrany dostępne w aplikacji**

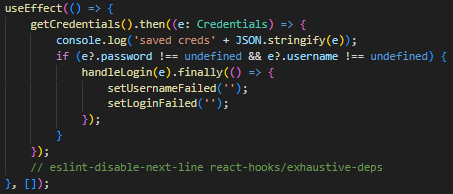
* Logowanie i rejestracja



Na ekranie po lewej stronie widać jest pola związane z rejestracją użytkownika, takie jak Username, Password, i Confirm Password. Poniżej znajduje się przycisk potwierdzjący rejestrację, i opcja do przełączenia na ekran logowania, który widać po prawej stronie. Na nim widnieją tylko pola do wpisania nazwy użytkownika   
i hasła, oraz przyciski do potwierdzenia logowania i przełączenia na ekran rejestracji.

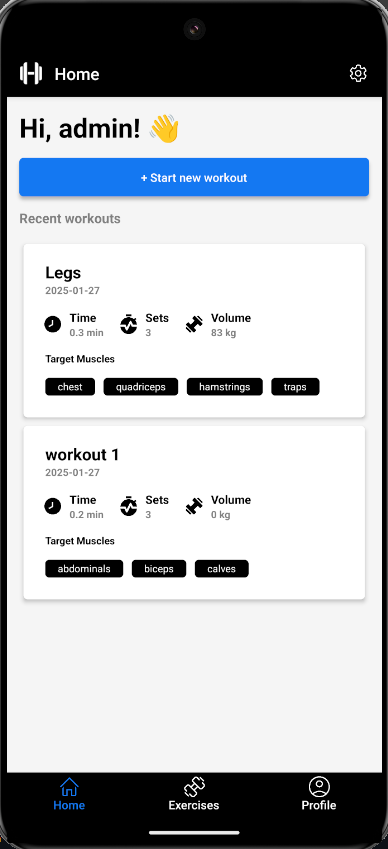
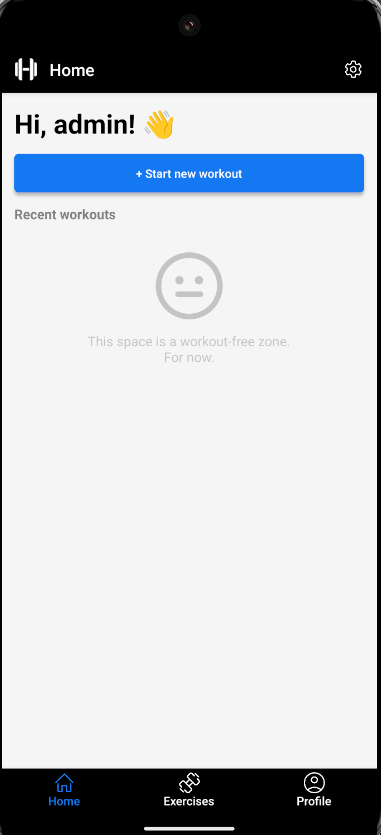


Funkcja autoryzująca logowanie użytkownika

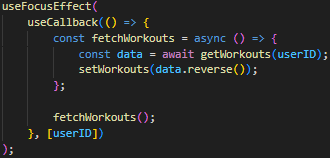


Funckja wykonująca autoamtyczne logowanie

* Ekran Główny

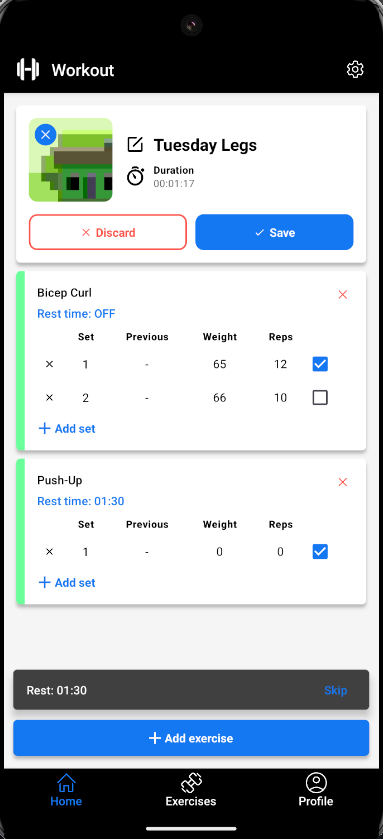
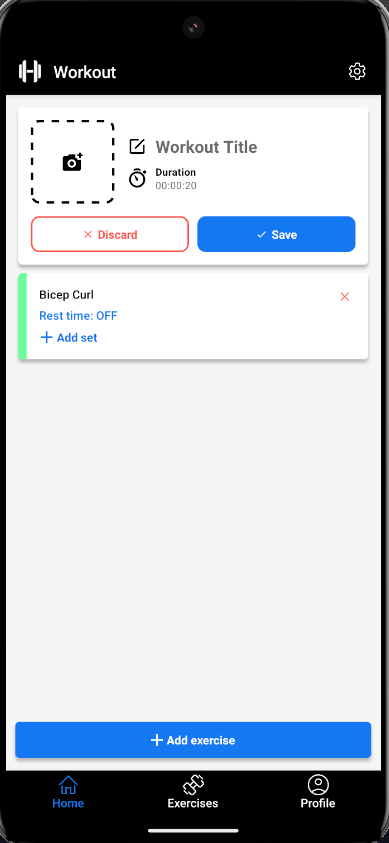


Po zalogowaniu się, aplikacja przenosi nas na ekran główny, gdzie znajdują się: przycisk przenoszący do ustawień w prawym górnym rogu, przycisk do rozpoczęcia treningu, dolna belka nawigacji, oraz jeśli istnieją, to lista poprzednich treningów.

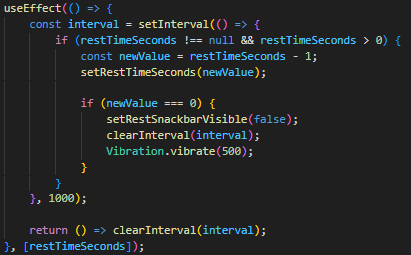


Funkcja pobierająca poprzednie treningi na podstawie id użytkownika

* Aktualny Trening



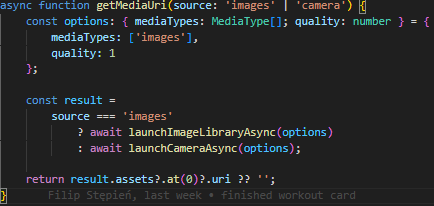
Na tym ekranie uzytkownik przeprowadza swój trening. W górnej części można dodać zdjęcie, ustawić tytuł, i zakończyć lub usunąć bieżący trening. Niżej wyświetlana jest lista ćwiczeń wykonywanych na tym treningu. Na karcie każdego ćwiczenia można ustawić czas odpoczynku, dodawać i usuwać serie, a w nich ustawiać ilośc powtórzeń oraz wagę. Gdy użytkownik zaznaczy serię jako wykonaną, na dole pojawia się powiadomienie odliczające czas odpoczynku, po którymn następują wibracje przypominające o kontynuowaniu treningu. Przycisk na samym dole służy do dodawania ćwiczeń do treningu. Ekran ten jest też używany do edycji wcześniej wykonanego treningu, dlatego jest to ekran   
z najbardziej złożonym kodem.



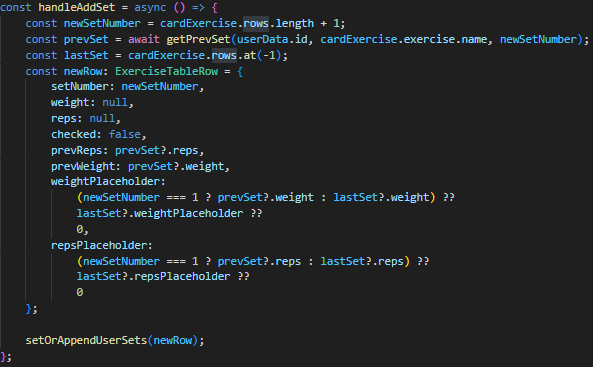
Funkcja odliczająca czas odpoczynku i wywołująca wibracje



Funkcja zapisująca wykonane ćwiczenia do API

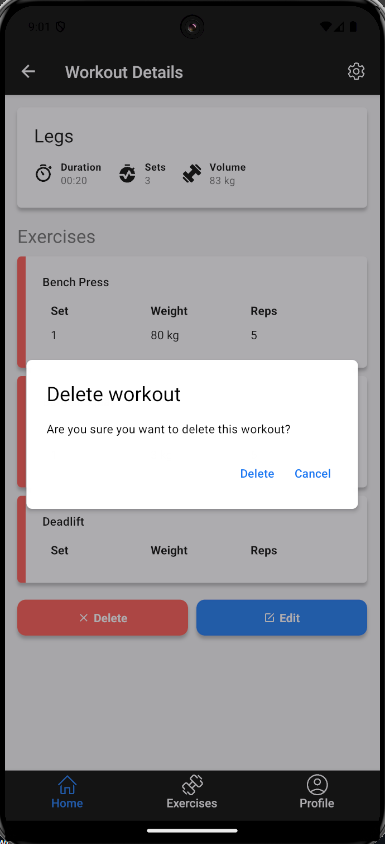
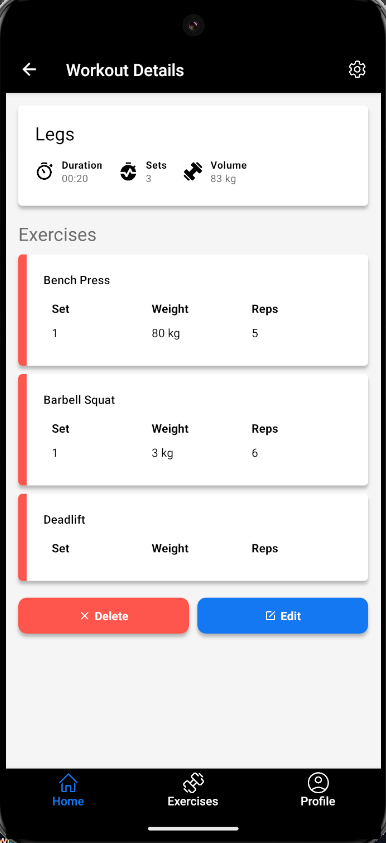


Funkcja pozwalająca zrobić zdjęcie lub wybrać zdjęcie z urządzenia

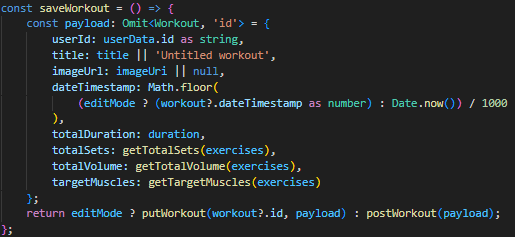


Funkcja obsługująca dodanie serii do danego ćwiczenia

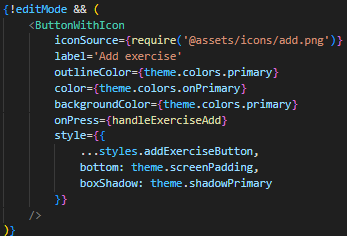
* Wykonany Trening



Ten ekran pozwala na podgląd szczegółów dotyczących treningu, posiada też przyciski do usuwania treningu oraz edytowania jego danych. Ten ekran jest bazowany na ekranie treningu.

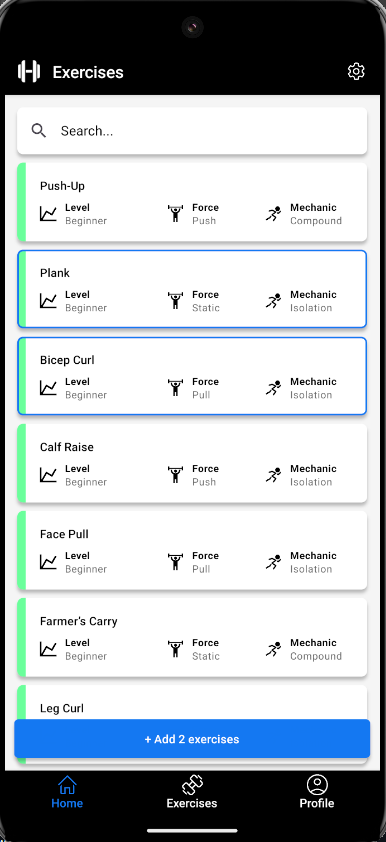
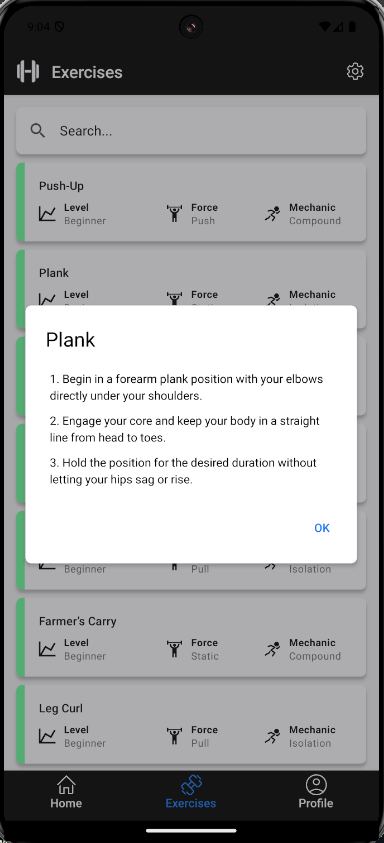
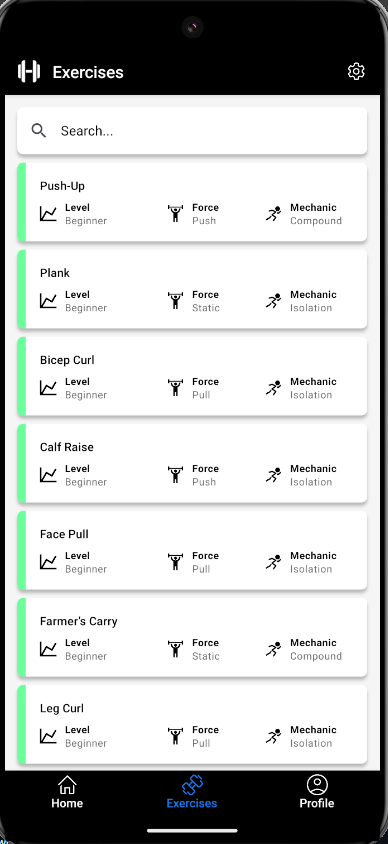


Funckja zapisująca trening w API

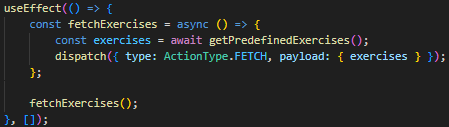


Przycisk wyświetlany w zależności od tego czy jesteśmy w trybie edycji danych treningu

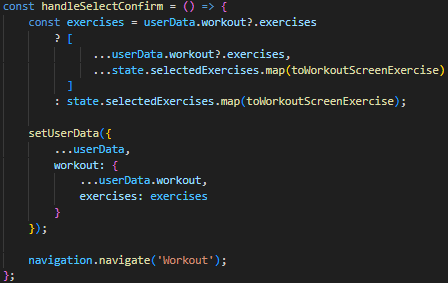
* Ćwiczenia



Ten ekran pozwala użytkownikowi na przeglądanie dostępnych ćwiczeń, oraz czytanie ich opisów/instrukcji. Po prawej stronie przedstawiona jest wersja tego ekranu, wchodząca w interakcje z ekranem treningu, gdzie użytkiownik wybiera ćwiczenia które chce wykonać, i po kliknięciu przycisku wraca na ekran treningu, gdzie ćwiczenia zostają dodane.

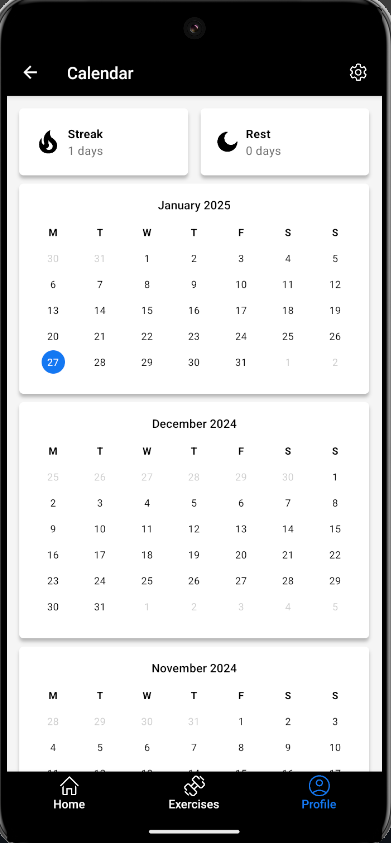
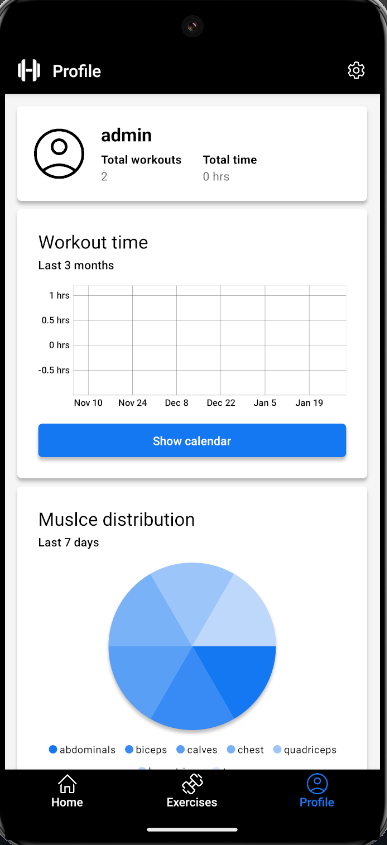


Funckja pobierająca ćwiczenia z API

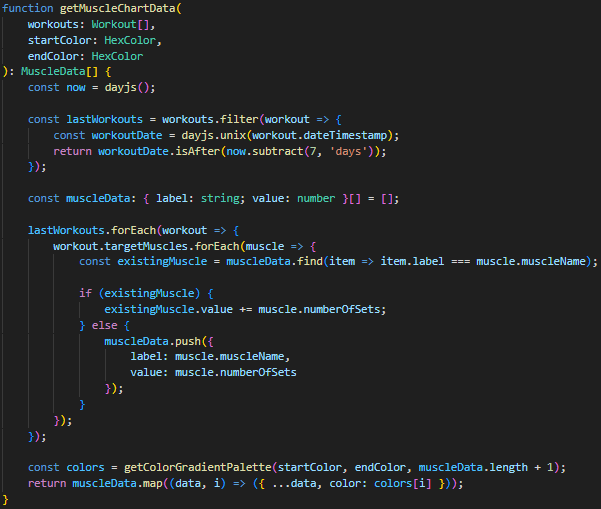


Funkcja obsługująca wybór ćwiczeń do treningu

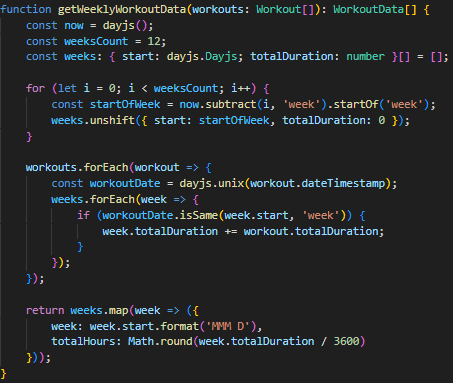
* Profil ze statystykami, kalendarz



Na profilu u samej góry znajdują się podstawowe dane użytkownika. Poniżej widać tygodniowy wykres treningów (w tym przypadku pusty). Pod nim znajduje się przycisk przenoszący na ekran kalendarzam gdzie można zobaczyć w jakie dni odbywało się trening. Pod Przyciskiem znajduje się wykres pokazujący rozłożenie obciążenia mięśni na treningach.

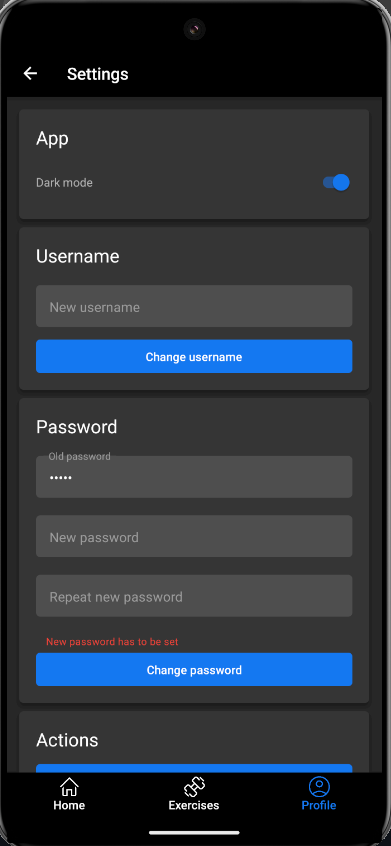
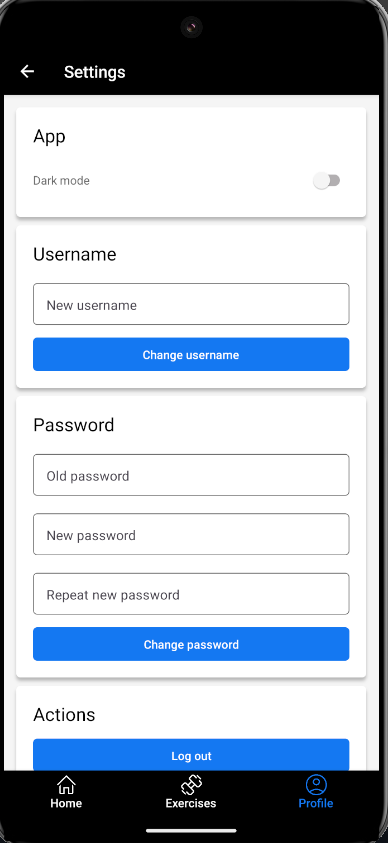


Funkcja pobierająca dane do wykresu mięśniowego\

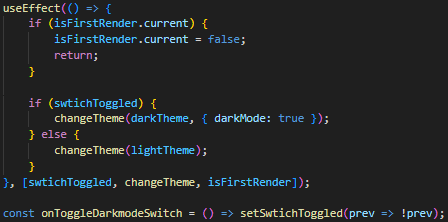


Funckja pobierająca i przypisująca dane do wykresu treningów

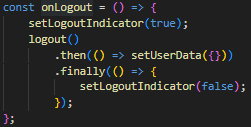
* Ustawienia



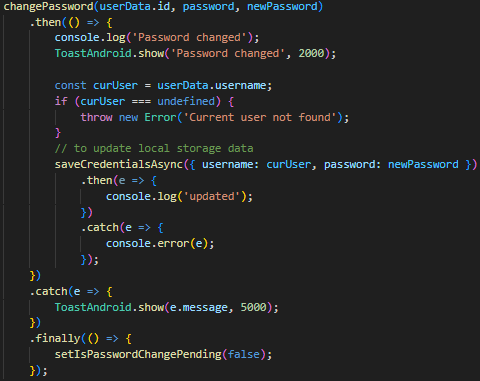
Na ekranie ustawień pierwsza możliwość to zmiana motywu aplikacji. Poniżej użytkownik może zmienić dane logowania (nazwę i hasło), i na samym dole jest możliwość wylogowania się z aplikacji. Po kliknięciu przycisku następuje wylogowanie i przeniesienie na ekran logowania.



Funkcja renderująca elementy aplikacji w zależności od wybranego motywu.



Funkcja obsługująca wylogowanie



Funkcja obsługująca zmianę hasła użytkownika

* 1. **Link do repozytorium**

Poniżej znajduje się link do zdalnego repozytorium gdzie można zobaczyć kod całej aplikacji:

<https://github.com/kurczakooo/React-Native-Project>

1. **Testy aplikacji**

|  |  |
| --- | --- |
| Wpisanie niepoprawnych danych logowania | Wpisanie pustych danych logowania |
| Błąd połączenia z API | Brak treningów do wyświetlenia |
| Brak statystyk do wyświetlenia (profil) | Brak statystyk do wyświetlenia (kalendarz) |
| Przekroczenie licznika odpoczynku (przed) | Przekroczenie licznika odpoczynku (po) |
| Błąd przy zmianie loginu | Niepoprawne powtórzone hasło |
| Niepoprawne hasło | Próba wprowadzenia pustego hasła |

1. **Podział Pracy**

Początkowy podział pracy przedstawia się następująco:

* **Filip Stępień** - odpowiedzialny za definicję typów, ekran ćwiczeń i ekran profilu użytkownika,
* **Rafał Grot** – odpowiedzialny za ekrany logowania, rejestracji, ustawień i API,
* **Damian Karwat** – odpowiedzialny za ekran główny, ekran aktualnie wykonywanego treningu, dolny pasek nawigacji.

W dalszej części pracy zadania poniekąd nakładały się na siebie, jako że np. ekrany korzystają ze wspólnego kontekstu, bądź wchodzą w interakcje ze sobą nawzajem.   
Z tego powodu przydzielano zadania dynamiczne podczas rozwiązywania kolejnych problemów w aplikacji.

1. **Podsumowanie**

W aplikacji z powodzeniem udała się implementacja wszystkich założeń   
i funkcjonalności, które były planowane na początku pracy. Podczas pracy nad projektem napotkano wiele wyzwań, takich jak :

* Integracja różnych funkcjonalności w obrębie wspólnego kontekstu aplikacji,
* Zapewnienie płynnego działania na różnych urządzeniach oraz optymalizacja interfejsu użytkownika,
* Przechowywanie danych w przemyślany, ustrukturyzowany sposób, tak aby posługiwanie się nimi było proste,
* Synchronizacja danych treningowych w taki sposób aby można było elastycznie zapisywać je do API, a następnie pobierać i edytować lub usuwać,
* Implementacja komunikacji i przekazywania sygnałów pomiędzy zagnieżdżonymi komponentami aplikacji.

Wszystkie te problemy zostały pomyślnie rozwiązane. Aplikacja była testowana zarówno na emulatorach, jak i rzeczywistych urządzeniach, aby upewnić się, że działa poprawnie i zapewnia spójne doświadczenie użytkownika.

* Odciążenie kontekstu aplikacji z dużej ilości danych, aby przyśpieszyć czasy ładowania,
* Kompatybilność aplikacji z wieloma urządzeniami (pomimo poprawnego działania, zdarza się że na różnych platformach elementy GUI zachowują się inaczej).

W przyszłości aplikacja mogłaby zostać rozszerzona o dodatkowe funkcje, takie jak:

* Możliwość logowania i tworzenia konta przez np. Google lub Facebook,
* Możliwość udostępniania postępów lub zdjęć w mediach społecznościowych,
* Synchronizacja z innymi aplikacjami, np. aplikacjami do diety,
* Rozszerzone statystyki spersonalizowane dla użytkowników,
* Bardziej motywujący system śledzący progres (np. w postaci nagród lub osiągnięć).