

Architektura Systemów Progresywnych w Inżynierii Aplikacji Treningowych: Synteza Fizjologii Wysiłku, Algorytmiki i Strategii Retencji Użytkownika

1. Wstęp: Ewaluacja Dokumentacji Źródłowej i Paradygmat Projektowy

Współczesny rynek aplikacji fitness, którego wartość szacowana jest na dziesiątki miliardów dolarów, odchodzi od prostych rejestratorów aktywności (loggers) na rzecz zaawansowanych systemów doradczych opartych na sztucznej inteligencji i dynamicznych algorytmach.¹ Kluczem do sukcesu w tym sektorze nie jest już samo gromadzenie danych, lecz ich interpretacja w celu optymalizacji procesu adaptacji biologicznej użytkownika. Niniejszy raport stanowi techniczną i merytoryczną analizę mechanizmów progresji w treningu siłowym, mającą na celu dostarczenie deweloperom kompletnego frameworku do budowy silnika treningowego.

W pierwszej fazie analizy dokonano krytycznej oceny dostarczonych materiałów źródłowych pod kątem ich przydatności w procesie wytwarzania oprogramowania (Software Development Life Cycle - SDLC).

1.1. Analiza Porównawcza i Rekomendacja Operacyjna

Dokonując dysekcji dwóch kluczowych dokumentów – "Progresja Obciążeń i Powtórzeń w Treningu.pdf"² oraz "Lista kontrolna przygotowania treści.pdf"² – w kontekście potrzeb architekta systemu i dewelopera backendu, można sformułować jednoznaczną hierarchię przydatności.

Dokument "Progresja Obciążeń i Powtórzeń w Treningu.pdf"² stanowi fundament logiczny i operacyjny dla silnika aplikacji. Jego struktura jest bezpośrednio przekładalna na kod źródłowy. Dokument ten definiuje:

- **Zmienne wejściowe i sterujące (Input/Control Variables):** Precyzyjnie rozróżnia intensywność, objętość i częstotliwość jako parametry funkcji adaptacyjnej.
- **Logikę warunkową (Conditional Logic):** Oferuje gotowe schematy decyzyjne typu „IF/THEN” (np. warunki inicjacji progresji, obsługa porażki treningowej), co jest niezbędne do zaprogramowania pętli zwrotnych w aplikacji.
- **Matematykę stosowaną:** Wskazuje konkretne modele matematyczne (formuły 1RM Epleya vs Brzyckiego) oraz mechanizmy „Plate Math” (matematyki talerzy), które są

krytyczne dla UX (User Experience) i wykonalności planu.

- **Taksonomię metod:** Klasyfikuje metody progresji (LP, DP, DDP) w sposób umożliwiający stworzenie modularnej architektury klas lub mikroserwisów obsługujących różne typy użytkowników.

Z kolei Dokument "Lista kontrolna przygotowania treści.pdf" ² pełni funkcję nadrzędnej warstwy walidacyjnej (Validation Layer) i compliance. Jest to zbiór wytycznych opartych na stanowiskach organizacji takich jak ACSM (American College of Sports Medicine) i NSCA (National Strength and Conditioning Association). Choć mniej przydatny przy pisaniu samego algorytmu progresji, jest kluczowy dla:

- **Bezpieczeństwa (Safety Constraints):** Definiuje granice bezpiecznych przyrostów (zasada „2 na 2”), co pozwala na implementację „bezpieczników” (hard stops) w algorytmie, zapobiegających generowaniu nierealistycznych lub niebezpiecznych obciążeń dla poczynających.
- **Onboardingu i Edukacji:** Dostarcza treści niezbędnych do wytłumaczenia użytkownikowi decyzji podjętych przez algorytm (np. dlaczego aplikacja nie zwiększyła ciężaru mimo udanej sesji).

Synteza dla Dewelopera: Architektura „Core Engine” aplikacji powinna zostać oparta na dynamicznych modelach z dokumentu ², natomiast parametry brzegowe, wartości domyślne dla nowych kont oraz moduły edukacyjne muszą być skalibrowane w oparciu o konserwatywne standardy z dokumentu.² Takie połączenie zapewnia balans między efektywnością treningową (Performance) a bezpieczeństwem (Safety).

2. Ontologia Danych Treningowych: Definicja Zmiennych i Stanów Użytkownika

Zanim przejdziemy do algorytmów, konieczne jest zdefiniowanie modelu danych. System treningowy nie operuje w próżni; musi rozumieć kontekst biomechaniczny i fizjologiczny każdej zmiennej. Prosta baza danych rejestrująca weight i reps jest niewystarczająca dla nowoczesnego systemu adaptacyjnego.

2.1. Zmienne Sterujące (Control Variables) i Ich Reprezentacja Cyfrowa

W inżynierii systemów treningowych wyróżniamy trzy główne wektory, którymi algorytm może manipulować w celu wywołania adaptacji, znanej jako Progresywne Przeciążenie (Progressive Overload).

2.1.1. Intensywność (Intensity)

W ujęciu fizjologicznym i systemowym, intensywność nie jest synonimem „trudności” (effort),

lecz miarą obciążenia zewnętrznego lub wewnętrznego.²

- **Absolutna (Load):** Wartość liczbową w kg/lbs. Wymaga normalizacji w bazie danych (np. przechowywanie wszystkiego w kg i konwersja w UI).
- **Relatywna (%1RM):** Procent ciężaru maksymalnego. Algorytm musi dynamicznie aktualizować e1RM (estimated 1-Rep Max) po każdej sesji, aby procenty odnosiły się do aktualnych możliwości użytkownika, a nie wartości historycznych sprzed miesiąca.
- **Subiektywna (RPE/RIR):** *Rate of Perceived Exertion* (skala 1-10) lub *Reps In Reserve*. Jest to kluczowa zmienna dla autoregulacji.² System musi traktować RPE jako *dane wejściowe użytkownika* (input), które korygują *dane wyjściowe systemu* (output - sugerowany ciężar).

2.1.2. Objętość (Volume) – Zmiana Paradygmatu

Tradycyjnie objętość w aplikacjach liczona była jako Volume Load (Tonaż): Serie × Powtórzenia × Ciężar.

Analiza Krytyczna: Nowoczesne badania nad hipertrofią 4 wskazują, że tonaż jest metryką ułomną. Wykonanie 100 powtórzeń z ciężarem 1 kg daje tonaż 100 kg, ale zerowy bodziec wzrostowy.

Implementacja Nowoczesna: Aplikacja powinna priorytetyzować „Liczbę Ciężkich Serii” (Hard Sets).

- Definicja Hard Set: Seria wykonana z zachowaniem 0-4 RIR (powtórzeń w zapasie).²
- **Wniosek dla kodu:** Algorytm zliczający objętość musi filtrować serie „rozgrzewkowe” (warm-up sets) i zliczać tylko „serie robocze” (working sets) spełniające kryterium intensywności (np. $RPE \geq 6$). Jest to zgodne z koncepcją „Effective Reps” – tylko ostatnie 5 powtórzeń przed upadkiem stymuluje włókna wysokoprogowe.⁴

2.1.3. Gęstość i Czas (Density & Tempo)

Często pomijana zmienna w prostych trackerach. Skrócenie czasu przerwy przy zachowaniu ciężaru jest formą progresji.⁵ Aplikacja powinna rejestrować rest_timer i korelować go z wydajnością. Jeśli użytkownik wykonał ten sam trening w 45 minut zamiast 60 minut, algorytm powinien to odnotować jako wzrost „Work Capacity”.

2.2. Stratyfikacja Użytkownika: Model Adaptacji a nie Czasu

Błąd kategoryzacji użytkownika jest najczęstszą przyczyną porażki aplikacji fitness (tzw. churn). Zgodnie z modelem Marka Rippetoe oraz analizą dokumentacji², użytkowników nie należy dzielić według czasu stażu (np. „trenuję rok”), lecz według **tempa adaptacji homeostatycznej (SRA Curve - Stress/Recovery/Adaptation)**.

Tabela 1: Klasyfikacja użytkownika w logice aplikacji

Klasa	Czas cyklu	Charakteryst	Zalecany	Logika
-------	------------	--------------	----------	--------

Użytkownika (User Class)	adaptacyjnego	tytuła Fizjologiczna	Model Progresji	Aplikacji (Backend)
Nowicjusz (Novice)	24-72 h (Sesja-do-Sesji)	Zdolność do regeneracji i superkompensacji przed kolejnym treningiem. „Efekt nowicjusza”.	Linear Progression (LP)	Next_Load = Current_Load + Increment. Ignoruj RPE w decyzjach o ciężarze; bazuj na binarnej logice Success/Fail w zaliczeniu powtórzeń.
Średniozaawansowany (Intermediate)	~1 tydzień (Weekly)	Stres pojedynczej sesji wymaga dłuższego czasu regeneracji. Progresja liniowa kończy się plateau.	Dynamic Double Progression (DDP), Wave Loading	Progresja następuje po zsumowaniu objętości tygodniowej lub zaliczeniu „top setu”. Wprowadzenie mikrocykli (Volume/Intensity days).
Zaawansowany (Advanced)	1 miesiąc+ (Block/Mesocycle)	Adaptacja wymaga kumulacji stresu przez wiele tygodni. Bliskość genetycznego potencjału.	Periodization (Block), Autoregulation	Złożone cykle: Akumulacja → Transmutacja → Realizacja (Peaking). Decyzje oparte na trendach długoterminowych i analizie danych.

Implikacja dla dewelopera: System musi posiadać zmienną stanu `adaptation_rate`, która jest dynamiczna. Jeśli użytkownik klasy „Novice” zalicza „Stall” (brak postępu) 3 razy z rzędu mimo resetu ciężaru, system musi automatycznie wywołać procedurę `upgrade_user_class()`

do poziomu Intermediate, zmieniając algorytm doboru obciążeń.²

3. Architektura Algorytmów Progresji: Implementacja Logiczna

W tej sekcji przedstawiono „serce” aplikacji – logikę sterującą doбором obciążeń. Każdy z poniższych modeli powinien być zaimplementowany jako oddzielna strategia w kodzie, możliwa do przypisania do konkretnego ćwiczenia.

3.1. Progresja Liniowa (Linear Progression - LP) – Silnik dla Nowicjuszy

Jest to model deterministyczny, najskuteczniejszy w pierwszych 3-9 miesiącach treningu. Zakłada stały wzrost intensywności (Load) przy stałej objętości (Sets/Reps).²

Algorytm "Standard LP":

Python

```
def calculate_next_load_lp(current_load, sets_completed, target_reps, increment_value):
    # Sprawdź, czy wszystkie serie zostały wykonane zgodnie z celem
    success = all(reps >= target_reps for reps in sets_completed)

    if success:
        return current_load + increment_value
    else:
        # Logika porażki (Fail)
        increase_fail_counter()
        if fail_count >= 3:
            return trigger_reset(current_load) # -10% ciężaru (patrz sekcja 5)
        else:
            return current_load # Powtórz ciężar w następnej sesji
```

Zasada „2 na 2” (NSCA/ACSM) – Bezpiecznik Systemowy:

Dokument 2 oraz wytyczne NSCA 8 promują zasadę „2-for-2 rule”.

- *Logika:* Jeśli w ostatniej serii (last set) użytkownik wykonał $\text{target_reps} + 2$ w dwóch kolejnych treningach, zwiększ ciężar.
- *Krytyka wdrożeniowa:* Z perspektywy retencji użytkownika w aplikacji, zasada ta jest **zbyt wolna** dla zdrowych nowicjuszy, którzy mogą dokładać ciężar co trening (co 48h).
- *Zastosowanie:* Należy ją zaimplementować jako domyślną strategię dla użytkowników

starszych (50+), po kontuzjach lub w ćwiczeniach izolowanych, gdzie szybki przyrost siły jest niemożliwy. Dla głównych bojów (Squat, Deadlift) u młodych adeptów, agresywna LP jest lepsza dla motywacji (dopamina).⁸

3.2. Podwójna Progresja (Double Progression - DP)

Model standardowy dla ćwiczeń akcesoryjnych (np. hantle). Manipuluje najpierw objętością (reps), potem intensywnością (load). Zapobiega to sytuacji, w której użytkownik musi zwiększyć ciężar hantla o 2 kg (co przy 10 kg hantlu stanowi skok o 20%!).²

Logika DP (Klasyczna):

Użytkownik otrzymuje zakres powtórzeń, np. 8-12.

1. Wybiera ciężar, z którym jest w stanie wykonać dolny zakres (np. 3x8).
2. Celem jest dojście do górnego zakresu we wszystkich seriach (3x12).
3. Może to zająć kilka treningów (np. T1: 10,9,8 -> T2: 11,10,9 -> T3: 12,12,12).
4. *Trigger*: Dopiero gdy $\text{Min}(\text{Reps_in_sets}) == \text{Upper_Limit}$, system sugeruje zwiększenie ciężaru i resetuje cel na 3x8.²

Wada (Sandbagging): Użytkownik może podświadomie oszczędzać siły w pierwszej serii (robiąc 10 zamiast 12), aby mieć pewność zaliczenia trzeciej serii. To prowadzi do suboptymalnego bodźca w najświeższym stanie układu nerwowego.

3.3. Dynamiczna Podwójna Progresja (Dynamic Double Progression - DDP) – Złoty Standard Hipertrofii

Model zoptymalizowany pod hipertrofię, zyskujący popularność w zaawansowanych aplikacjach. Traktuje każdą serię jako „niezależną jednostkę progresji”.²

Logika DDP:

Założenie: Zakres 8-12 powtórzeń. Każda seria żyje własnym życiem.

- Set 1: Użytkownik wykonał 12 powtórzeń (górny limit). -> **Decyzja:** Zwiększ ciężar dla Set 1 w następnym treningu.
- Set 2: Użytkownik wykonał 9 powtórzeń (środek zakresu). -> **Decyzja:** Utrzymaj ciężar dla Set 2 w następnym treningu.
- Set 3: Użytkownik wykonał 8 powtórzeń (dolny limit). -> **Decyzja:** Utrzymaj ciężar.

Implikacje dla bazy danych: Struktura danych musi pamiętać historię obciążenia per seria, a nie per ćwiczenie.

W kolejnym treningu UI aplikacji musi wyświetlić:

- Set 1: 22.5 kg (Progresja)
- Set 2: 20 kg (Brak zmian)
- Set 3: 20 kg (Brak zmian)

Przewaga nad DP: DDP pozwala na szybszą progresję w pierwszej serii, nie hamując jej

zmęczeniem skumulowanym w trzeciej serii. Badania sugerują, że jest to skuteczniejsze dla hipertrofii, ponieważ maksymalizuje napięcie mechaniczne (mechanical tension) w stanie świeżości.¹²

4. Matematyka Obciążenia i Mikro-progresja: Precyzja Systemu

Aplikacja obliczeniowa operuje na liczbach rzeczywistych, ale rzeczywistość siłowni jest dyskretna (talerze, hantle). System musi mostkować tę przepaść.

4.1. Formuły Szacowania 1RM: Dobór Kontekstowy

Estymacja 1RM (One Rep Max) jest kluczowa dla grywalizacji i planowania obciążeń. Jednakże, uniwersalne stosowanie jednego wzoru jest błędem. Analiza literatury¹⁵ wykazuje różną dokładność w zależności od liczby powtórzeń.

Wzór Epleya:

$$1RM = W \cdot (1 + \frac{r}{30})$$

- *Zastosowanie:* Wyższa dokładność dla niskich zakresów powtórzeń (1-5 reps). W wysokich zakresach (10+) ma tendencję do znacznego zawyżania wyniku, co może prowadzić do przepisania użytkownikowi zbyt dużego ciężaru w nowym bloku.¹⁵

Wzór Brzyckiego:

$$1RM = W \cdot \frac{36}{37 - r}$$

- *Zastosowanie:* Bardziej konserwatywny. Często preferowany w badaniach naukowych. Mniej podatny na wykładnicze błędy przy wyższych powtórzeniach, choć powyżej 10 powtórzeń jego precyzja również spada.¹⁵

Wzór Wathena:

Zalecany przez niektóre źródła dla zakresów 10+ powtórzeń, stosowany w zaawansowanej analityce.¹⁹

Rekomendacja Hybrydowa dla Algorytmu:

Zamiast sztywnego wyboru, aplikacja powinna stosować funkcję warunkową w backendzie:

Python

```
def estimate_1rm_hybrid(weight, reps):
    if reps <= 5:
        return epley_formula(weight, reps) # Dokładniejszy w strefie siły
    elif reps <= 10:
        return brzycki_formula(weight, reps) # Standardowy kompromis
    else:
        return wathen_formula(weight, reps) # Lepszy dla wytrzymałości siłowej
```

Takie podejście minimalizuje błąd estymacji („Edge Case” dla wysokich powtórzeń), co jest krytyczne dla bezpieczeństwa użytkownika przy przechodzeniu z fazy hipertrofii do siły.¹⁶

4.2. Mikro-obciążenia (Microloading) i Logika „Plate Math”

Jednym z największych problemów algorytmicznych jest „ściana” wynikająca z fizycznych ograniczeń sprzętu.

- Przysiad 100 kg: skok o 2.5 kg to 2.5% → Akceptowalne.
- Wyciskanie OHP 30 kg: skok o 2.5 kg to 8.3% → Często niemożliwe do wykonania (Fail).

Rozwiązanie Systemowe – Moduł „Microloading Calculator”:

1. **Kontekst Ćwiczenia:** Algorytm musi wiedzieć, czy ćwiczenie jest złożone (Squat) czy izolowane (Lateral Raise). Dla izolacji *zakazuje* liniowej progresji wagowej, chyba że użytkownik posiada talerze ułamkowe (fractional plates).²
2. **Ścieżka Alternatywna:** Jeśli skok wagi jest niemożliwy (brak sprzętu lub zbyt duży % skoku), system automatycznie przełącza się na **Progresję Objętościową** na danym ciężarze.
 - *Scenariusz:* Użytkownik utknął na 3x5 @ 30kg. Następny hantel to 32.5kg.
 - *Akcja Algorytmu:* Zmień cel na 3x6 @ 30kg, potem 3x7, aż szacowane 1RM pozwoli na bezpieczny skok na 32.5kg.²¹

To podejście, oparte na badaniach nad rekrutacją jednostek motorycznych²¹, jest kluczowe dla utrzymania motywacji użytkownika, który w przeciwnym razie zniechęciłby się ciężkimi porażkami.

5. Autoregulacja i Pętle Zwrotne (Feedback Loops)

Współczesny system treningowy musi być responsywny (Responsive Training). Sztywne plany nie uwzględniają stresu życiowego, jakości snu czy gorszej dyspozycji.

5.1. RPE vs. RIR: Problem Wiarygodności Danych

- **RPE (Rate of Perceived Exertion):** Skala 1-10.

- **RIR (Reps In Reserve):** Powtórzenia w zapasie (RPE 9 = 1 RIR).

Badania 22 wskazują na istotny problem: Początkujący (Novice) mają niską zdolność do trafnej oceny RPE. Często zaniżają ocenę wysiłku, kończąc serię na rzeczywistym RIR 5-6, raportując RPE 9.

Implikacja Algorytmiczna: System nie powinien ufać RPE w przypadku użytkowników oznaczonych jako „Novice”. Dla tej grupy priorytetem jest prosta, binarna progresja (zaliczone/niezaliczone). RPE powinno być zbierane w tle (shadow data collection) w celach kalibracyjnych, ale nie powinno sterować obciążeniem. Dopiero dla użytkowników „Intermediate” i „Advanced”, RPE staje się wiarygodnym inputem do korekty ciężaru (Autoregulated Load Adjustment).³

5.2. Velocity Based Training (VBT) – Obiektywizacja

Dla aplikacji współpracujących z sensorami (np. akcelerometry w smartwatchach), VBT stanowi „ultimate truth”.²⁴

- *Logika:* Jeśli prędkość (velocity) w pierwszej serii spadnie o >20% względem średniej historycznej dla tego ciężaru, oznacza to skumulowane zmęczenie neuronowe.
- *Akcja:* Automatyczna redukcja ciężaru o 5-10% na daną sesję (Autoregulated Drop). Jest to znacznie precyzyjniejsze niż subiektywne RPE.

6. Zarządzanie Zmęczeniem: Algorytmy Deloadu i Detekcji Plateau

System, który tylko dodaje ciężar, nieuchronnie doprowadzi użytkownika do kontuzji lub wypalenia (Overtraining). Inteligentna aplikacja musi zarządzać „krzywą zmęczenia”.

6.1. Detekcja Plateau i Reset

Algorytm musi monitorować wskaźnik Stall_Count dla każdego ćwiczenia.

- *Warunek:* Brak progresji (wzrostu Load lub Reps) przez 3 kolejne sesje.
- *Akcja: Reset.* Zmniejsz ciężar roboczy o 10-15% (np. z 100 kg na 85-90 kg) i wymuś ponowną, powolną wspinaczkę. Pozwala to na regenerację tkanki łącznej i przełamanie bariery psychicznej.²

6.2. Strategie Deloadu: Volume vs. Intensity

Istnieją dwie szkoły implementacji deloadu w aplikacjach ²⁶:

1. **Proaktywny (Periodic):** Szttywno zaplanowany co 4-6 tygodni. Bezpieczny, ale może przerwać „flow” użytkownika.
2. **Reaktywny (Autoregulated):** Inicjowany na podstawie symptomów (spadek siły, chronicznie wysokie RPE).

Krytyczna Korekta Merytoryczna:

Wiele aplikacji błędnie implementuje deload poprzez zmniejszenie ciężaru. Badania 28 jednoznacznie wskazują, że utrzymanie intensywności (ciężaru) przy jednoczesnej redukcji objętości (liczby serii o 30-50%) jest lepsze dla zachowania adaptacji siłowych i techniki.

- *Zły Deload*: 3 serie x 10 powtórzeń @ 50% ciężaru (zbyt lekkie, detrening).
- *Dobry Deload (Algorytmiczny)*: 1-2 serie x 10 powtórzeń @ 90% ciężaru (zachowanie bodźca nerwowego, redukcja stresu metabolicznego i systemowego).

6.3. Volume Landmarks (Punkty Orientacyjne Objętości)

Wdrożenie modelu Mike'a Israetela (MEV/MAV/MRV) ³¹ pozwala na długofalową periodyzację.

- **MEV (Minimum Effective Volume)**: Punkt startowy bloku (np. 10 serii/tydzień).
- **Progresja Objętościowa**: Algorytm dodaje 1-2 serie co tydzień, przesuwając użytkownika przez strefę MAV (Maximum Adaptive Volume).
- **MRV (Maximum Recoverable Volume)**: Granica możliwości regeneracyjnych (np. 20+ serii). Po jej osiągnięciu (wykrytym przez spadek wydajności), system wymusza deload.

Jest to zaawansowana logika, wymagająca od użytkownika raportowania wskaźników takich jak „Soreness” (obolałość) czy „Pump” w ankietach potreningowych, co pozwala systemowi na personalizację objętości per grupa mięśniowa.³²

7. Wnioski i Wytyczne Implementacyjne

Analiza materiałów ² oraz szerokiej bazy badań prowadzi do sformułowania następujących filarów dla deweloperów aplikacji fitness:

1. **Hierarchia Logiki**: Dokument ² stanowi podstawę dla backendu (silnika obliczeniowego), podczas gdy ² definiuje warstwę bezpieczeństwa i edukacji (frontend/interfejs).
2. **Indywidualizacja przez Algorytm**: Aplikacja musi dynamicznie przełączać modele progresji (z LP na DDP) w oparciu o wykryte tempo adaptacji użytkownika, a nie jego staż kalendarzowy.
3. **Precyzja Matematyczna**: Implementacja hybrydowych wzorów 1RM oraz logiki „Microloading/Plate Math” jest niezbędna, aby uniknąć frustracji użytkowników zderzających się z niemożliwymi skokami obciążeń.
4. **Inteligentny Deload**: Redukcja objętości przy zachowaniu intensywności jest jedyną naukowo uzasadnioną metodą deloadu, którą należy zakodować jako domyślną procedurę regeneracyjną.
5. **DDP jako Standard**: Dynamiczna Podwójna Progresja powinna być domyślnym modelem dla celów sylwetkowych (hipertrofia), oferując lepszą granularność postępów niż klasyczne modele liniowe.

Tabela 2: Matryca Implementacji Algorytmów Progresji

Cecha Algorytmu	Progresja Liniowa (LP)	Podwójna Progresja (DP)	Dynamiczna Podwójna Progresja (DDP)	Autoregulacja (RPE/RIR)
Główny Trigger	Ukończenie wszystkich serii zgodnie z celem	Ukończenie górnego zakresu powtórzeń we wszystkich seriach	Ukończenie górnego zakresu w <i>pojedynczej</i> serii	RPE < Cel (zapas siły)
Zmienna Manipulowana	Ciężar (Load)	Powtórzenia, następnie Ciężar	Ciężar (per seria)	Ciężar (Dnia/Sesji)
Grupa Docelowa	Nowicjusz (Novice)	Średniozaawansowany (izolacje)	Średniozaawansowany / Zaawansowany	Zaawansowany
Ryzyko Stagnacji	Wysokie (szybka ściana)	Średnie (sandbagging)	Niskie (ciągły bodziec)	Niskie (zależy od uczciwości)
Złożoność Kodu	Niska $O(1)$	Średnia $O(n)$	Wysoka $O(n*m)$ (tracking per set)	Wysoka (wymaga kalibracji modelu)

Zastosowanie powyższych wytycznych pozwoli na przekształcenie aplikacji z pasywnego dziennika treningowego w aktywne, inteligentne narzędzie sterujące procesem biologicznym użytkownika, co bezpośrednio przełoży się na wyniki treningowe i wskaźniki retencji (LTV).

Cytowane prace

1. Health & fitness tracker app insights for 2023 - Adjust, otwierano: stycznia 13, 2026, <https://www.adjust.com/blog/health-tracker-installs-and-retention-data/>
2. Progresja Obciążeń i Powtórzeń w Treningu.pdf
3. RPE vs. Percentage 1RM Loading in Periodized Programs Matched for Sets and

- Repetitions - Frontiers, otwierano: stycznia 13, 2026,
<https://www.frontiersin.org/journals/physiology/articles/10.3389/fphys.2018.00247/full>
4. Effective Reps vs. Volume: Stop Tracking Junk Work - Strive Workout Log, otwierano: stycznia 13, 2026,
<https://strive-workout.com/2026/01/02/effective-reps-vs-volume-hypertrophy-guide/>
 5. The Ten Rules of Progressive Overload - Bret Contreras, otwierano: stycznia 13, 2026, <https://bretcontreras.com/progressive-overload/>
 6. Practical Programming for Strength Training - ICDST E-print archive of engineering and scientific PDF documents, otwierano: stycznia 13, 2026,
<https://dl.icdst.org/pdfs/files/c225e2fa042cc55070a31c6d636e4f20.pdf>
 7. Practical Programming for Strength Training 3rd Edition Review - PowerliftingToWin, otwierano: stycznia 13, 2026,
<https://www.powerliftingtowin.com/practical-programming-review/>
 8. Total Force Wellness Column: Guidelines to progress your physical training over time, otwierano: stycznia 13, 2026,
<https://www.mycg.uscg.mil/News/Article/2910464/total-force-wellness-column-guidelines-to-progress-your-physical-training-over/>
 9. Testing and Profiling Athletes: Recommendations for Test Selection, Implementation, and Maximizing Information - NSCA, otwierano: stycznia 13, 2026,
<https://www.nsca.com/contentassets/d34fb088b444495da825298cb03526a8/testing-and-profile--weakley-et-al.pdf>
 10. RPE and RIR: The Complete Guide - MASS Research Review, otwierano: stycznia 13, 2026,
<https://massresearchreview.com/2023/05/22/rpe-and-rir-the-complete-guide/>
 11. The Double Progression Method for Getting Bigger and Stronger - Legion Athletics, otwierano: stycznia 13, 2026,
<https://legionathletics.com/double-progression/>
 12. Progression systems and their effect on your training: straight sets vs double progression : r/naturalbodybuilding - Reddit, otwierano: stycznia 13, 2026,
https://www.reddit.com/r/naturalbodybuilding/comments/1fgjmbn/progression_systems_and_their_effect_on_your/
 13. The Utility of Dynamic/Autoregulated Double Progression - JPS Health & Fitness, otwierano: stycznia 13, 2026,
<https://www.jpshealthandfitness.com.au/the-utility-of-dynamic-autoregulated-double-progression/>
 14. Progressive overload without progressing load? The effects of load or repetition progression on muscular adaptations - PubMed, otwierano: stycznia 13, 2026,
<https://pubmed.ncbi.nlm.nih.gov/36199287/>
 15. One-repetition maximum - Wikipedia, otwierano: stycznia 13, 2026,
https://en.wikipedia.org/wiki/One-repetition_maximum
 16. Do 1RM calculation formulae start to become inaccurate as the rep counts go way up?, otwierano: stycznia 13, 2026,

- <https://fitness.stackexchange.com/questions/46784/do-1rm-calculation-formulae-start-to-become-inaccurate-as-the-rep-counts-go-way>
17. Accuracy of Predicting One-Repetition Maximum from Submaximal Velocity in The Barbell Back Squat and Bench Press - PMC - NIH, otwierano: styczeń 13, 2026, <https://pmc.ncbi.nlm.nih.gov/articles/PMC9465738/>
 18. otwierano: styczeń 13, 2026, [https://pmc.ncbi.nlm.nih.gov/articles/PMC9465738/#:~:text=DiStasio%20\(2014\)%20reported%20that%20the,1993%3B%20DiStasio%2C%202014\).](https://pmc.ncbi.nlm.nih.gov/articles/PMC9465738/#:~:text=DiStasio%20(2014)%20reported%20that%20the,1993%3B%20DiStasio%2C%202014).)
 19. One-Rep Max Formulas Showdown / Volodymyr Agafonkin - Observable Notebooks, otwierano: styczeń 13, 2026, <https://observablehq.com/@mourner/one-rep-max-formulas-showdown>
 20. The Science of Microloading: Can Small Weight Increases Make a Big Difference?, otwierano: styczeń 13, 2026, <https://speediance.eu/blogs/news/the-science-of-microloading-how-small-weight-increases-lead-to-big-strength-gain>
 21. The Power of 2.5lb Plates: Your Secret Weapon for Sustainable Strength - Back 40 Fitness, otwierano: styczeń 13, 2026, <https://back40fitness.com/blogs/back-40-fitness/the-power-of-2-5lb-plates-your-secret-weapon-for-sustainable-strength>
 22. Ability to predict repetitions to momentary failure is not perfectly accurate, though improves with resistance training experience - PMC - NIH, otwierano: styczeń 13, 2026, <https://pmc.ncbi.nlm.nih.gov/articles/PMC5712461/>
 23. RIR vs RPE: The Clear Intensity Framework Adult Athletes Need - AFT Fitness Coaching, otwierano: styczeń 13, 2026, <https://aftfitnesscoaching.com/blogs/fitness-resources-blog/rir-vs-rpe-the-clear-intensity-framework-adult-athletes-need>
 24. The Resistance Training Dose Response: Meta-Regressions Exploring the Effects of Weekly Volume and Frequency on Muscle Hypertrophy and Strength Gains - Fisiología del Ejercicio, otwierano: styczeń 13, 2026, <https://www.fisiologiadelejercicio.com/wp-content/uploads/2025/12/The-Resistance-Training-Dose-Response.pdf>
 25. Progressive overload: the ultimate guide - GymAware, otwierano: styczeń 13, 2026, <https://gymaware.com/progressive-overload-the-ultimate-guide/>
 26. "You can't shoot another bullet until you've reloaded the gun": Coaches' perceptions, practices and experiences of deloading in strength and physique sports - Frontiers, otwierano: styczeń 13, 2026, <https://www.frontiersin.org/journals/sports-and-active-living/articles/10.3389/fspor.2022.1073223/full>
 27. Are Deloads Overrated? - Empowerlift | Powerlifting Coaching And Fitness Training, otwierano: styczeń 13, 2026, <https://empowerlifttraining.com/are-deloading-overrated/>
 28. otwierano: styczeń 13, 2026, https://levelsprotein.com/blogs/training/deload-week#:~:text=Evidence%20suggests%20that%20in%20strength,for%20reversing%20fatigue%5B*%5D.
 29. A Practical Approach to Deloading: Recommendations and Considerations for

Strength and Physique Sports - ResearchGate, otwierano: stycznia 13, 2026,
https://www.researchgate.net/publication/391802156_A_Practical_Approach_to_Deloiding_Recommendations_and_Considerations_for_Strength_and_Physique_Sports

30. Deloading Practices in Strength and Physique Sports: A Cross-sectional Survey - PMC - NIH, otwierano: stycznia 13, 2026,
<https://pmc.ncbi.nlm.nih.gov/articles/PMC10948666/>
31. Dr. Mike Israetel's Training Tips for Hypertrophy : r/weightroom - Reddit, otwierano: stycznia 13, 2026,
https://www.reddit.com/r/weightroom/comments/6674a4/dr_mike_israetels_training_tips_for_hypertrophy/
32. Set Volume for Muscle Size: The Ultimate Evidence Based Bible - Weightology, otwierano: stycznia 13, 2026,
<https://weightology.net/the-members-area/evidence-based-guides/set-volume-for-muscle-size-the-ultimate-evidence-based-bible/>