

Softwarebeschreibung – Workout Tracker

Kurzbeschreibung

Der Workout Tracker ist eine Web-Applikation, mit der Nutzer:innen ihre Trainings (Workouts) planen, aufzeichnen und auswerten können. Die Anwendung bietet eine einfache Erfassung von Übungen, Sätzen, Wiederholungen, Gewichten sowie Notizen. Fortschritte werden in Diagrammen visualisiert.

Zielgruppen und Stakeholder

- **Endnutzer:innen (Athlet:innen/Fitness-Interessierte):** Workouts erfassen, Fortschritt verfolgen
- **Trainer:innen (optional):** Trainingspläne bereitstellen, Feedback geben
- **Produkt/Team:** Betrieb, Weiterentwicklung, Qualitätssicherung

Hauptfunktionen (Scope)

- **Authentifizierung/Autorisierung:** Registrierung, Login, Passwort-Reset (JWT-basiert)
- **Workout-Verwaltung:** Erstellen, Lesen, Aktualisieren, Löschen (CRUD) von Workouts und Übungen
- **Vorlagen:** Wiederverwendbare Trainingspläne
- **Verlauf & Analytics:** Verlauf pro Übung, persönliche Bestleistungen, einfache Charts
- **Export:** CSV/JSON-Export der Workouts

Nicht-funktionale Anforderungen (Auszug)

- **Sicherheit/Privacy:** Passwort-Hashing, TLS, least privilege, DSGVO/Datensparsamkeit
- **Performance:** P95 API-Antwortzeit 500 ms bei typischer Last
- **Zuverlässigkeit:** Fehlerrobustheit, geordnete Migrationsstrategie
- **Usability:** Mobile-First, barrierearme UI

Systemkontext & Architekturüberblick

Die Lösung ist als klassische Web-App mit REST-API entworfen. Datenpersistenz über eine relationale Datenbank. Authentifizierung via JWT.

flowchart LR

```
User((Nutzer:in)) --> UI[Web App]
UI --> API[REST API (Node.js/TypeScript)]
API --> DB[(PostgreSQL)]
API --> Auth[(JWT / Password Hashing)]
API --> Notify[(Notification/Email, optional)]
```

Datenobjekte (grob)

- **User:** id, email, password_hash, created_at
- **Workout:** id, user_id, date, notes
- **Exercise:** id, name, category
- **Set:** id, workout_id, exercise_id, reps, weight, rpe

Abgrenzung

- Kein Social-Feed, keine Community-Features
- Kein komplexes Coaching-Portal (nur optionales Minimal-Sharing)

Risiken & Annahmen

- Annahme: Einzel-User-Fokus (später Teams erweiterbar)
- Risiko: Datenschutzverletzungen → starke Sicherheits- und Logging-Kontrollen nötig
- Risiko: Fehlende Testdatenqualität → definierte Seed-Daten für Tests

PDF-Export (Beispiel):

```
pandoc /Users/isaaclins/Documents/github/m450/03/aufgabe1-beschreibung.md \
-o /Users/isaaclins/Documents/github/m450/03/aufgabe1-beschreibung.pdf
```