

Prototyping Projektdokumentation

Name: Constantin Ochsner

E-Mail: ochsncon@students.zhaw.ch

URL der deployten Anwendung: <https://workoutpt.netlify.app/>

1. Einleitung

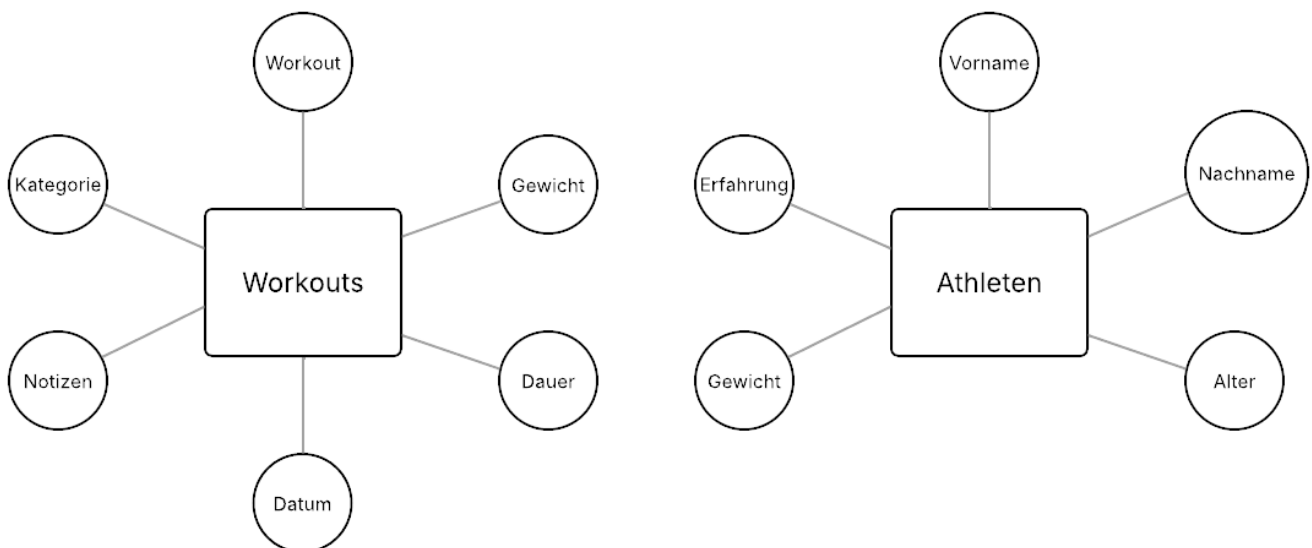
Der Name der Anwendung lautet «Workout Tracker» und dient der strukturierten Erfassung und Verwaltung von Trainingseinheiten. Ziel der App ist es, den Nutzer eine Möglichkeit zu bieten, ihre Workouts digital zu dokumentieren und bei Bedarf zu aktualisieren oder zu löschen. Im Zentrum der Anwendung steht die Verwaltung von Workouts, die über ein Formular hinzugefügt werden können. Jedes Workout enthält Informationen wie Workout, Gewicht, Dauer, Datum, Notizen sowie eine zugeordnete Kategorie (z. B. Kraft, Ausdauer oder Kombination). Eine Übersichtsliste zeigt alle erfassten Workouts an. Zusätzlich steht eine Live-Suchfunktion zur Verfügung, mit der gezielt nach dem Workout gefiltert werden kann.

Neben den Workouts können Athletenprofile eingesehen werden. Diese enthalten Basisdaten wie Name, Alter, Gewicht und Erfahrungsstufe. Die Athletenansicht dient ausschliesslich der Information.

Grundfunktionen:

- Startseite mit Karussell und thematischen Bildern
- Auflistung aller erfassten Workouts mit Bearbeitungs- und Löschfunktion
- Workouts erfassen mit Workout, Gewicht, Dauer, Datum, Notizen und Kategorie
- Athletenprofil-Ansicht (View Only)
- Navigation zwischen den Seiten mit Bootstrap-kompatiblem Layout
- Datenverwaltung via MongoDB mit SvelteKit und Deployment über Netlify

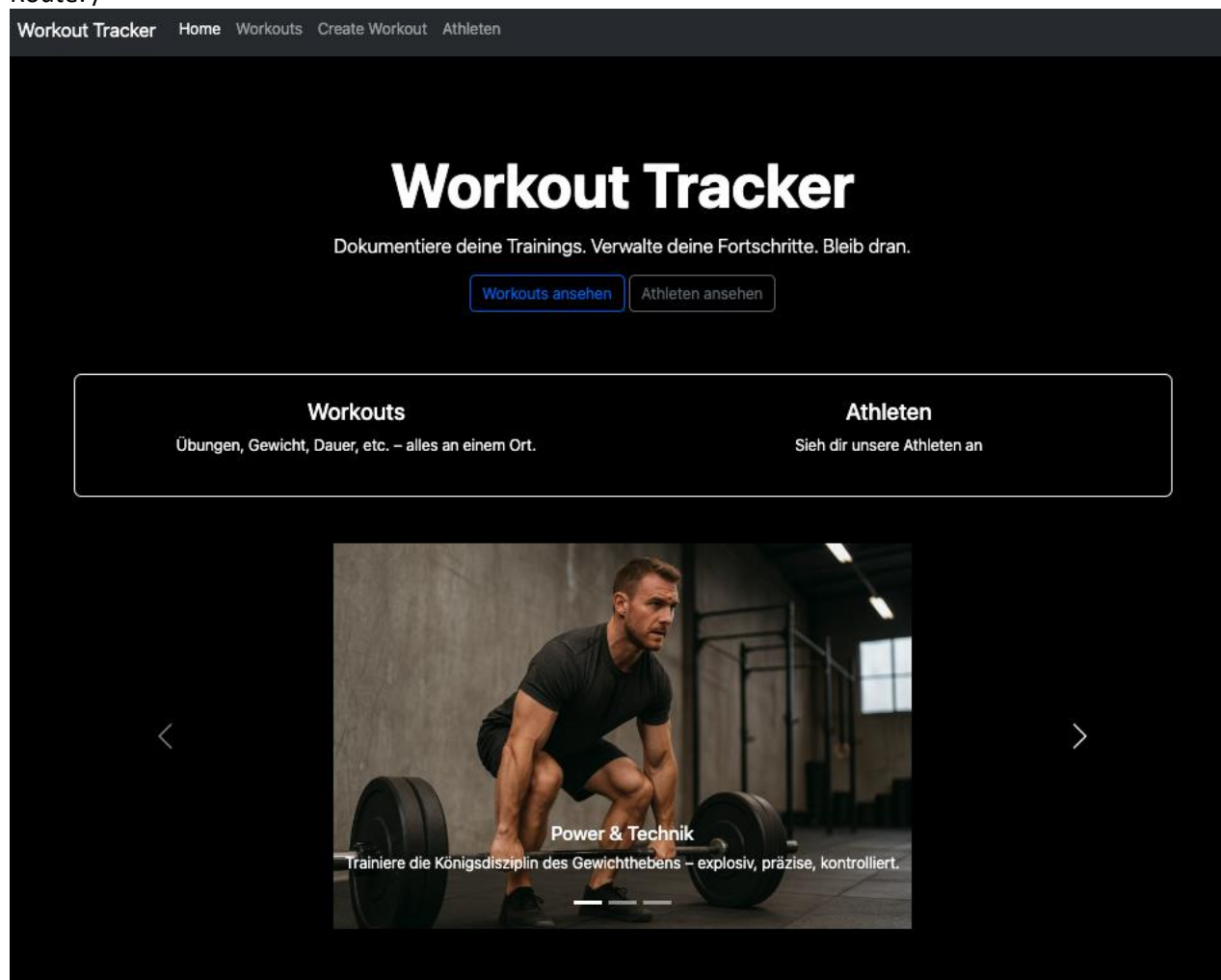
2. Datenmodell



3. Beschreibung der Anwendung

3.1. Startseite

Route: /



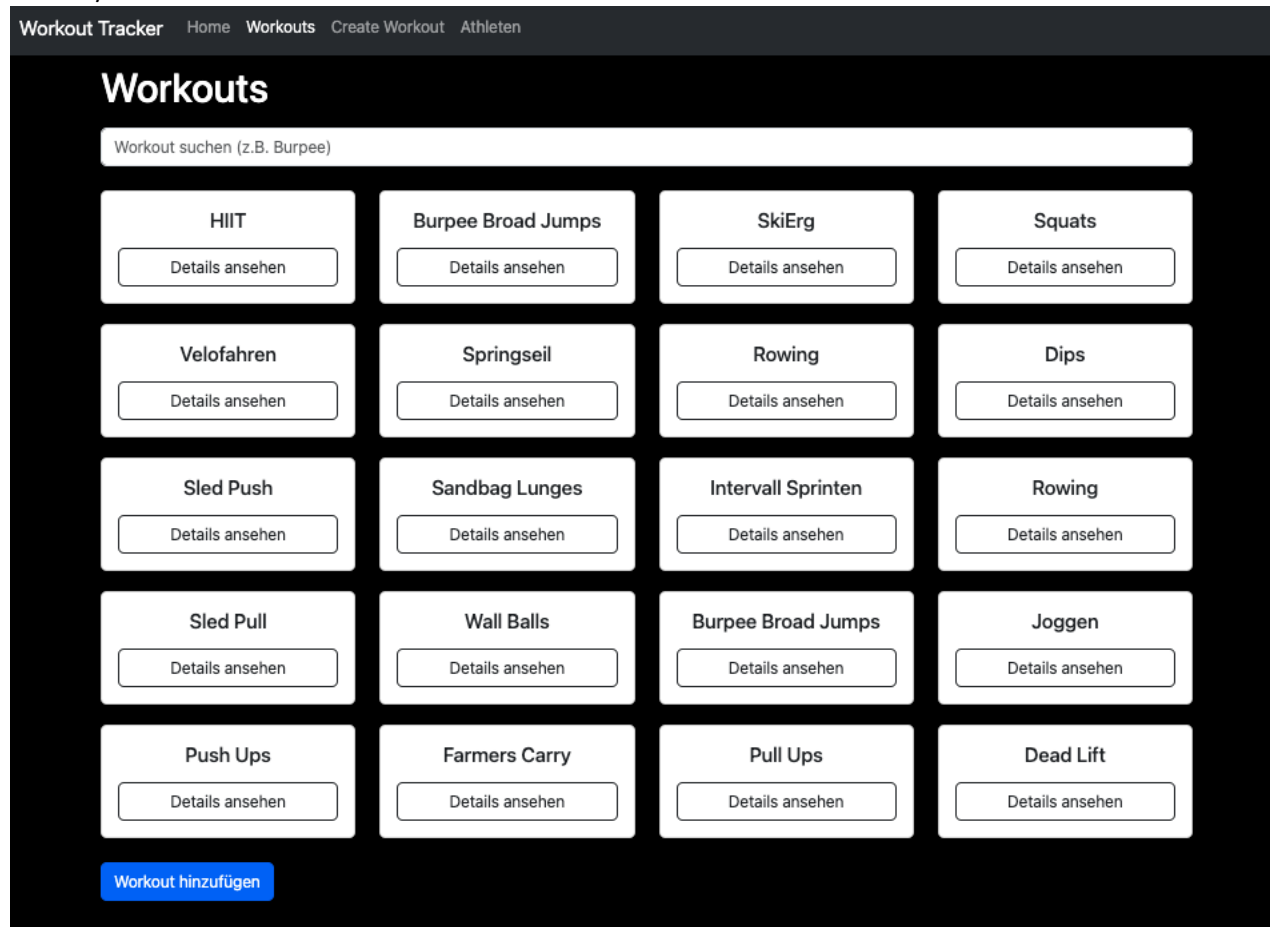
Auf der Startseite werden die zentralen Funktionen der Anwendung vorgestellt. In einem oberen Bereich werden ein Begrüßungstitel sowie zwei Buttons angezeigt, über die die Benutzer direkt zur Workout- oder Athletenübersicht navigieren können. Unterhalb davon werden die Vorteile der Anwendung mit Text erläutert. Im unteren Bereich ist ein Karussell mit Bildern zu typischen Kraftübungen eingebunden. Die Seite ist im Dark Mode gestaltet und nutzt Bootstrap-Komponenten für die Struktur und das Styling. Das Layout ist responsiv und damit auch auf mobilen Geräten nutzbar.

Dateien:

- routes/+page.svelte
- routes/+.layout.svelte

3.2. Workouts

Route: /workouts



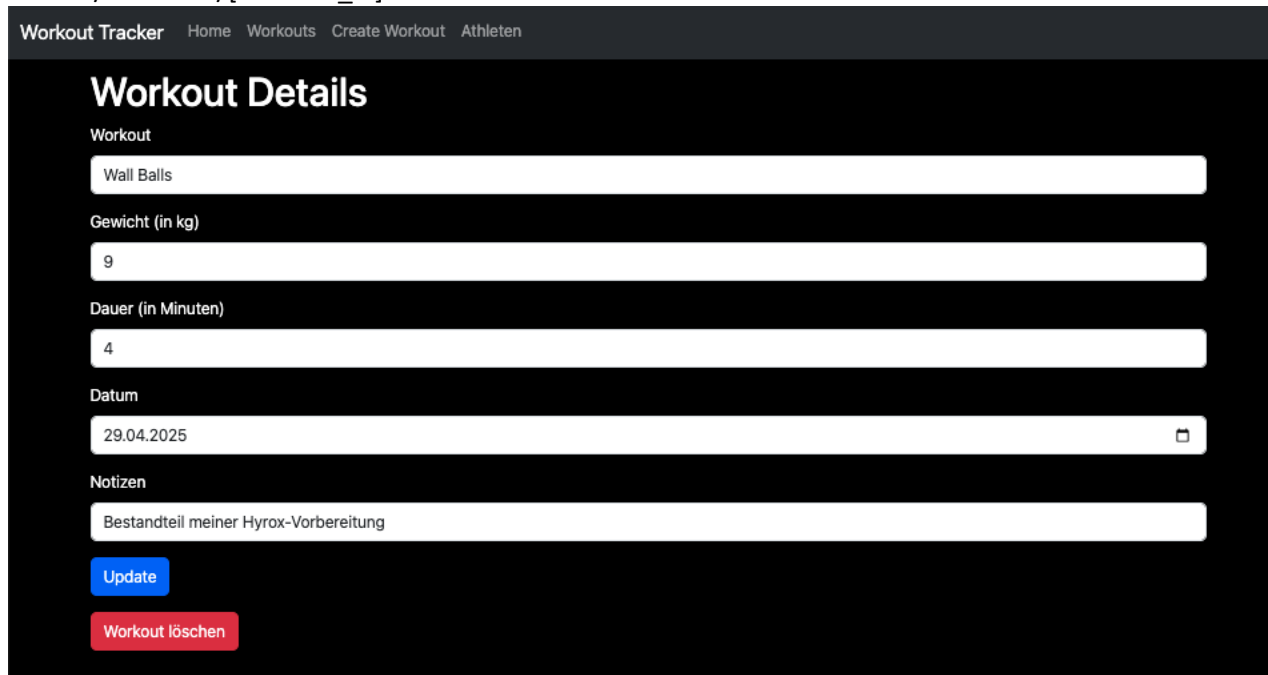
Auf dieser Seite werden alle erfassten Workouts als Karten dargestellt. Für die Darstellung der einzelnen Workouts wird die Komponente `Workouts.svelte` verwendet. Zu jedem Workout gibt es einen Button „Details ansehen“, über den man zur Detailansicht und Bearbeitung gelangt. In der Suchfunktion kann der Nutzer mit dem Namen nach dem Workout filtern. Mit dem Button «Workout hinzufügen» wird man zu «Create Workout» weitergeleitet. Die Workouts werden mittels der Klasse `db.js` abgerufen.

Dateien:

- `routes/workouts/+page.svelte`
- `routes/workouts/+page.server.js`
- `lib/components/Workouts.svelte`
- `db.js`

3.3. Workout Details

Route: /workouts/[workout_id]



Workout Tracker Home Workouts Create Workout Athleten

Workout Details

Workout

Gewicht (in kg)

Dauer (in Minuten)

Datum

Notizen

Update

Workout löschen

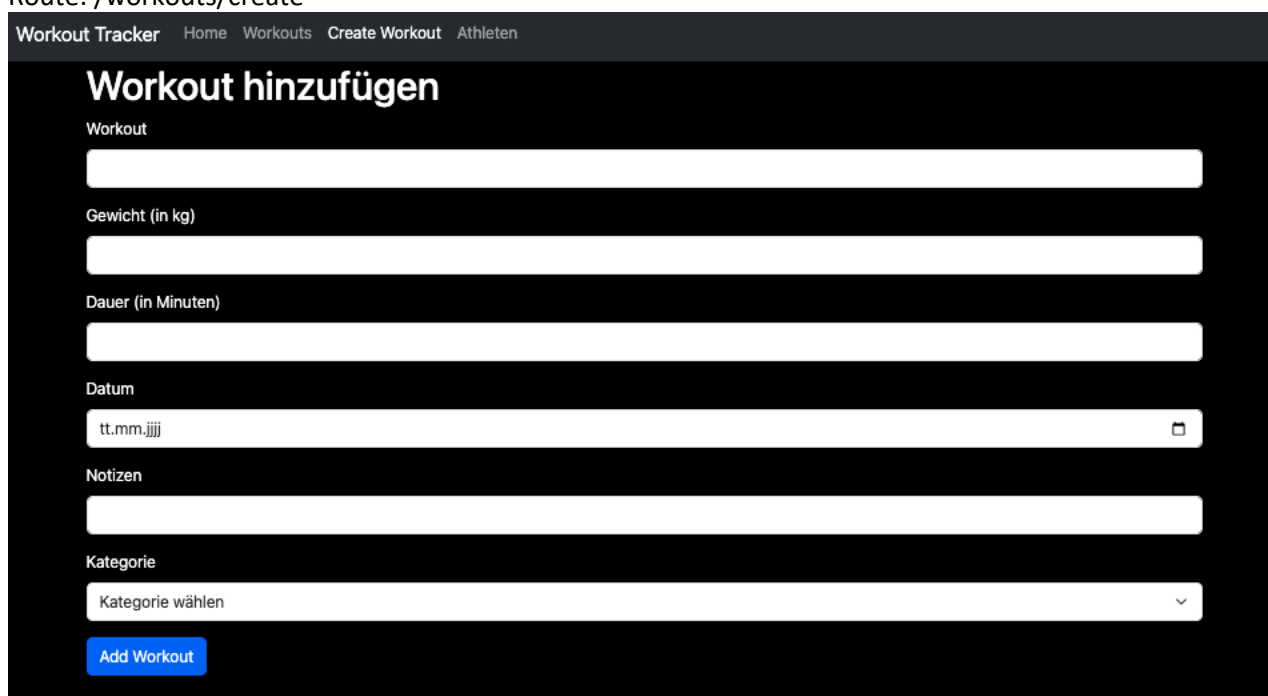
Auf dieser Seite werden die Details eines einzelnen Workouts angezeigt. Die Benutzer können das Workout direkt bearbeiten oder löschen. Die Felder: Workout, Gewicht, Dauer, Datum und Notizen sind dabei vorausgefüllt und können angepasst werden. Änderungen werden über den Button „Update“ gespeichert. Mit dem Button „Workout löschen“ wird der Eintrag aus der Datenbank im Mongo DB entfernt und die Benutzer zurück zur Übersicht weitergeleitet. Die Daten werden serverseitig über die Datei +page.server.js geladen und verarbeitet. Die Formularverarbeitung erfolgt ebenfalls serverseitig über definierte update- und delete-Actions.

Dateien:

- routes/workouts/[workout_id]/+page.svelte
- routes/workouts/[workout_id]/+page.server.js

3.4. Workout hinzufügen

Route: /workouts/create



Workout Tracker Home Workouts Create Workout Athleten

Workout hinzufügen

Workout

Gewicht (in kg)

Dauer (in Minuten)

Datum

Notizen

Kategorie

Add Workout

Auf dieser Seite kann ein neues Workout erfasst werden. Über ein Formular werden Informationen wie Workout, Gewicht, Dauer, Datum, Notizen und Kategorie eingegeben. Die Eingaben werden beim Absenden über die create-Action an die MongoDB-Datenbank übermittelt und dort gespeichert. Nach erfolgreicher Eingabe wird eine Bestätigung ausgegeben, welche nach drei Sekunden wieder verschwindet.

Dateien:

- routes/workouts/create/+page.svelte
- routes/workouts/create/+page.server.js

3.5. Athleten

Route: /athleten

Workout Tracker Home Workouts Create Workout Athleten				
Athleten				
Vorname	Nachname	Alter	Gewicht (in kg)	Erfahrung
Clare	Dellenbach	37	59.8	Profi
Elsie	McBain	78	49.6	Anfänger
Buddy	Scanderet	38	119.9	Fortgeschrittene
Filmer	Normant	21	57.1	Profi
Tarah	Simoncini	57	63.2	Anfänger
Jeffy	Driver	21	66	Profi
Angie	Theis	76	50	Anfänger
Shirline	O'Glessane	19	103.4	Fortgeschrittene
Sadella	Brooking	36	58.6	Anfänger
Wendell	Orta	76	69.9	Profi
Clary	Croisdall	75	43.1	Fortgeschrittene
Casie	Gerger	20	67.4	Fortgeschrittene
Charo	Krabbe	79	96.9	Anfänger
Jillie	Gurnett	33	78.9	Profi
Lauree	Fibbitts	62	62.8	Fortgeschrittene
Dallas	Wisam	73	105	Anfänger
Niki	Chalcroft	74	81.1	Fortgeschrittene
Oliy	Parram	54	106.6	Profi
Kenn	Seary	17	113.3	Profi
Crysta	Davsley	46	86.5	Anfänger

Auf dieser Seite werden alle registrierten Athleten in Tabellenform aufgelistet. Die Tabelle enthält die Spalten Vorname, Nachname, Alter, Gewicht und Erfahrungslevel. Eine Bearbeitungsfunktion ist nicht vorgesehen – die Athletenansicht ist read-only. Die Tabelle reagiert auf Benutzerinteraktion: Beim Überfahren einer Zeile mit der Maus wird diese visuell hervorgehoben (Hover-Effekt). Die Daten werden serverseitig geladen und direkt in der Seite eingebunden.

Dateien:

- routes/athleten/+page.svelte
- routes/athleten/+page.server.js

4. Erweiterungen

Im Rahmen der Umsetzung wurden vier funktionale Erweiterungen realisiert, die über die Grundanforderungen hinausgehen:

4.1. Redirect nach dem Löschen eines Workouts

Auf der Seite `/workouts/[workout_id]` wurde in der Datei `+page.server.js` eine Weiterleitungsfunktion eingebaut. Nach dem Löschen eines Workouts wird der Benutzer automatisch zur Workout-Übersichtsseite `/workouts` zurückgeleitet. Dadurch entfällt die Notwendigkeit für eine manuelle Navigation nach dem Löschvorgang.

4.2. Visuelles Feedback in der Navigationsleiste

In der Datei `+layout.svelte` wurde die Navigation um ein Feature erweitert, das visuell kennzeichnet, welche Seite aktuell aktiv ist. Der entsprechende Menüpunkt wird hervorgehoben.

4.3. Suchfunktion bei den Workouts

In der Datei `+page.svelte` der Route `/workouts` wurde oberhalb der Workout-Liste eine Suchfunktion implementiert. Über ein Eingabefeld kann nach Workouts gefiltert werden. Bereits während der Eingabe wird die Workout-Liste in Echtzeit angepasst, wodurch nur noch passende Einträge angezeigt werden.

4.4. Erfolgsmeldung mit Timeout

Nach dem erfolgreichen Aktualisieren oder Hinzufügen eines Workouts erscheint eine Rückmeldung in Form der Nachricht „Workout aktualisiert!“ bzw. «Workout hinzugefügt!». Zur Verbesserung der Benutzerfreundlichkeit wird die Meldung nicht dauerhaft angezeigt, sondern verschwindet automatisch nach drei Sekunden.