**Studienarbeit zur Anwendung „GymJourney“**

Alexej Kunz (00193829)

Hochschule für Angewandte Wissenschaften Hof

Moderne App- und Webentwicklung

Sommersemester 2023

23.06.2023

Inhalt

[Abbildungsverzeichnis 3](#_Toc133446442)

[1 Einführung 4](#_Toc133446443)

[1.1 Anforderungsanalyse für GymJourney 4](#_Toc133446444)

[1.2 Architektur und Entwurf 5](#_Toc133446445)

[2 Implementierung 6](#_Toc133446446)

[3 Walkthrough durch die GymJourney Website 6](#_Toc133446447)

[4 Fazit 8](#_Toc133446448)

[Literaturverzeichnis 8](#_Toc133446449)

# Abbildungsverzeichnis

[1 React.js Logo 5](#_Toc133441818)

[2 Express.js Logo 5](#_Toc133441818)

[3 MySQL Logo 5](#_Toc133441818)

[4 Technologie-Stack 6](#_Toc133441818)

[5 Login-Seite 7](#_Toc133441818)

[6 Register-Seite 7](#_Toc133441818)

[7 Home-Seite 8](#_Toc133441818)

8 [Trainingsplan erstellen 8](#_Toc133441818)

9 [Der erstellte Trainingsplan 8](#_Toc133441818)

10 [Trainingsplan bearbeiten 8](#_Toc133441818)

11 [Der bearbeitete Trainingsplan 8](#_Toc133441818)

12 [Trainingsplan löschen 8](#_Toc133441818)

13 [Erfolgreiche Löschung eines Trainingsplans 8](#_Toc133441818)

14 [Workouts-Seite 8](#_Toc133441818)

15 [Workouts-Seite mit gespeicherten Workouts 8](#_Toc133441818)

16 [Übungen-Seite 8](#_Toc133441818)

17 [Übungen-Seite mit gespeicherter Übung 8](#_Toc133441818)

18 [Informationen zu einer Übung 8](#_Toc133441818)

19 [Fehlerseite 8](#_Toc133441818)

# Einführung

Die Website GymJourney richtet sich an sportbegeisterte Personen. Sie soll ihnen dabei helfen, möglichst einfach und unkompliziert die Übersicht über ihre Trainingspläne zu behalten. Im Folgenden wird eine Anforderungsanalyse für diese Website durchgeführt.

## Anforderungsanalyse für GymJourney

//Todo: Evtl kurze Einführung in Anforderungsanalyse und warum man diese macht, vlt mit Quelle, evlt weitere Anforderungen durch Anforderungen der Studienarbeit ableiten

* Bei der Anwendung soll es sich um eine Single Page Application (SPA) handeln
* Die Anwendung soll es dem Nutzer ermöglichen, einen Account zu erstellen und sich mit diesem Account anzumelden
* Die Anwendung soll eine sichere Aufbewahrung des Passworts des Nutzers gewährleisten, indem das Passwort zumindest gehasht und gesalted wird
* Die Sitzung des angemeldeten Nutzers soll über Cookies verwaltet werden
* Der Nutzer soll sich über einen Button von der Anwendung ausloggen können
* Es sollen nur angemeldete Nutzer Zugriff auf die Anwendung haben
* Der Nutzer soll in der Lage sein eigene Trainingspläne, Trainingseinheiten (auch Workouts genannt), und Übungen zu erstellen, angezeigt zu bekommen, zu bearbeiten und zu löschen
* Die Anwendung soll bei den Übungen YouTube-Videos einbinden können, sodass der Nutzer diese durch einen Link abspeichern kann und auf der Website anschauen kann
* Die Anwendung soll eine Fehlerseite anzeigen, sollte eine Route nicht gefunden werden
* Die Anwendung soll responsive sein und somit sowohl auf mobilen Endgeräten als auch auf Desktop-Geräten erfolgreich dargestellt werden können
* Die Anwendung soll nach der Hierarchiereihenfolge Trainingspläne, Trainingseinheiten und Übungen aufgebaut sein – wird ein Trainingsplan gelöscht so sollen auch alle Trainingseinheiten und Übungen dieses Trainingsplans gelöscht werden
* Der Nutzer soll nur auf seine eigenen Daten zugreifen können – Zugriffe auf Daten anderer Nutzer sollen vermieden werden
* Die Anwendung soll lediglich im Dark Mode gestaltet werden

## Architektur und Entwurf

In diesem Abschnitt soll die Architektur der Anwendung erläutert werden, wobei zunächst auf den eingesetzten Technologie-Stack eingegangen wird.

Das Frontend der Anwendung wurde mit React.js geschrieben. React.js ist ein JavaScript-Framework, das verwendet wird, um Benutzerschnittstellen zu erstellen. Es ist das beliebteste komponentenbasierte JavaScript-Framework [vgl. 1]. Alternativen hierzu wären unter anderem Angular.js, Vue.js oder Hof.js, jedoch wurde sich aufgrund persönlicher Vorerfahrung mit React.js für React.js entschieden. Für das Backend wurde Node.js mit dem Framework Express verwendet. Node.js an sich ermöglicht es unter anderem serverseitige Applikationen mit JavaScript zu schreiben. Node.js hat von sich aus jedoch nicht die Möglichkeit HTTP-Anfragen wie GET, PUT, POST, DELETE… zu unterstützen, weshalb Express genutzt wird, da es sonst nötig wäre, eine eigene Logik für das Verarbeiten von HTTP-Anfragen zu schreiben. Express ist das beliebteste Node-Webframework [vgl. 2]. Alternative Backend Frameworks wären unter anderem Laravel oder Ruby on Rails, jedoch wurde auch hier aufgrund von persönlichen Vorerfahrungen mit JavaScript für Express entschieden. Um Daten aus dieser Anwendung zu speichern, wird eine Datenbank benötigt. Hier wurde sich für MySQL entschieden, da man hier relativ einfach die geforderten voneinander abhängigen Entitäten abbilden kann und es eines der beliebtesten Datenbankmanagementsysteme ist, was wohl vor Allem daran liegt, dass es häufig in WordPress-Anwendungen zum Einsatz kommt [vgl. 3]. Mögliche Alternative wären hier unter anderem PostgreSQL, SQLite oder auch MariaDB. Eine Übersicht über den Technologie-Stack bietet Abbildung 4.

|  |  |  |
| --- | --- | --- |
| Frontend | Backend | Datenbank |
| React.js  Abbildung 1: React.js Logo [4] | Ein Bild, das Logo enthält.  Automatisch generierte BeschreibungExpress.js  Abbildung 2: Express.js Logo [5] | MySQL  Abbildung 3: MySQL Logo [6] |

Abbildung 4: Technologie-Stack

//TODO: Eventuell folgendes hinzufügen: Diagramm, das die verschiedenen Komponenten der Seite und die Beziehung zueinander zeigt, z.B. mit Flussdiagramm oder Mind Map. Auch Datenbankstruktur könnte man erklären (auch mit hashen und salten der Passwörter).

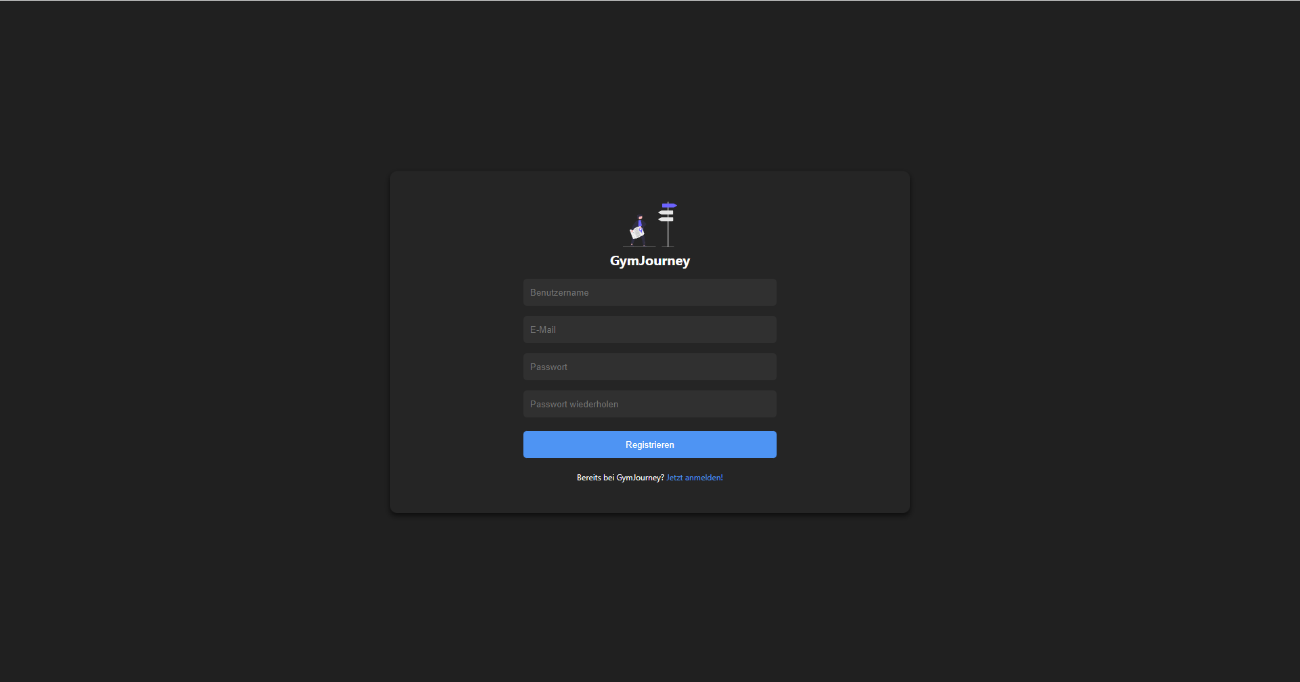
# Implementierung

# Walkthrough durch die GymJourney Website

Im Folgenden wird durch die GymJourney-Website geleitet, um die Bedienung sowie die Möglichkeiten der Website zu dokumentieren.

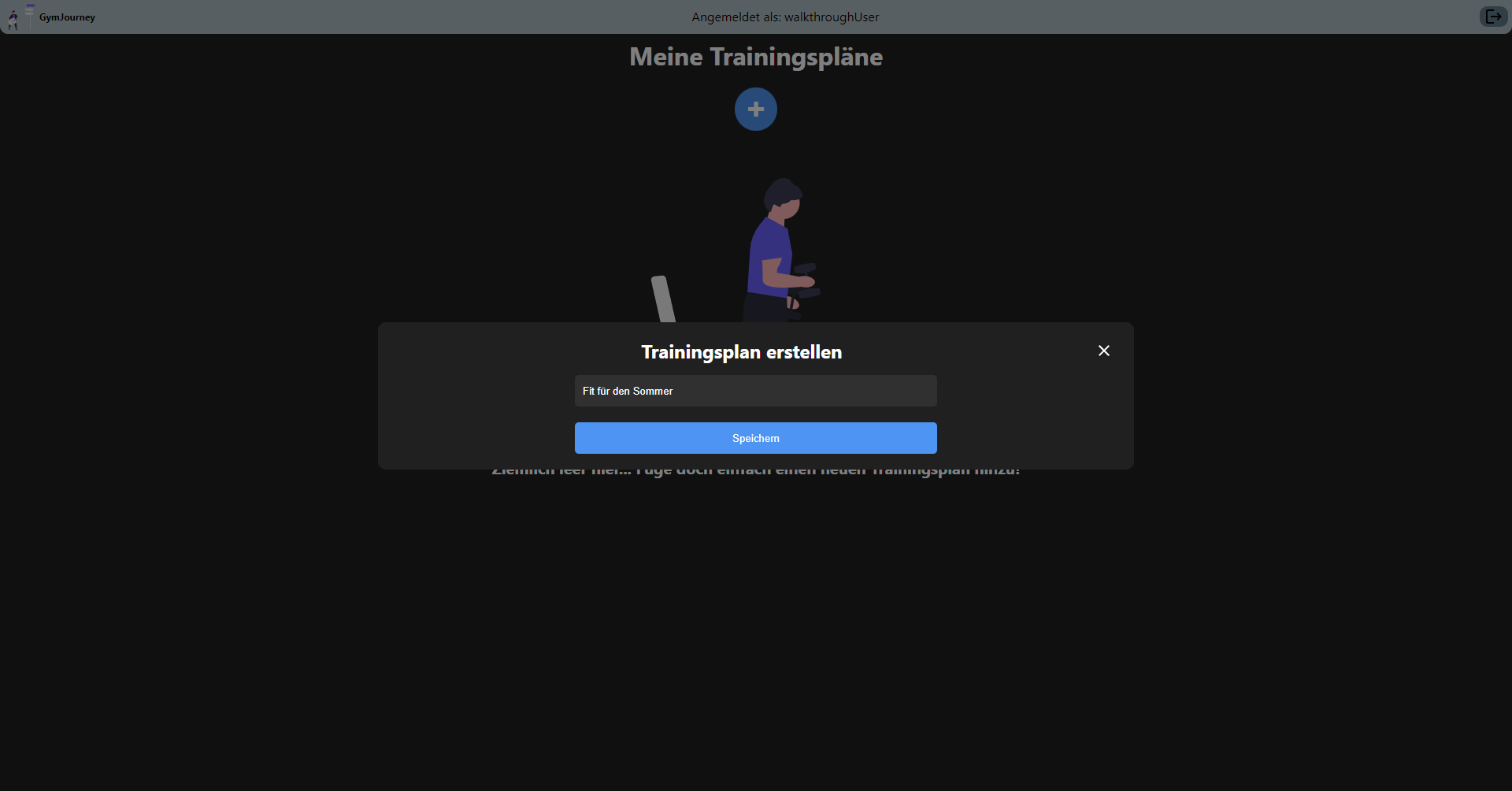
Zunächst landet der Nutzer auf der in Abbildung 5 dargestellten Login-Seite, zumindest wenn er nicht angemeldet ist. Grund dafür ist, dass der Nutzer für jede Funktionalität, die diese Website bietet, angemeldet sein muss. Sollte der Nutzer noch kein Account besitzen, so kann er auf den Link „Jetzt registrieren“ klicken und landet auf der Register-Seite, um sich einen neuen Account zu erstellen. Diese ist in Abbildung 6 abgebildet.

Ein Bild, das Text, Bildschirm, Dunkel enthält.

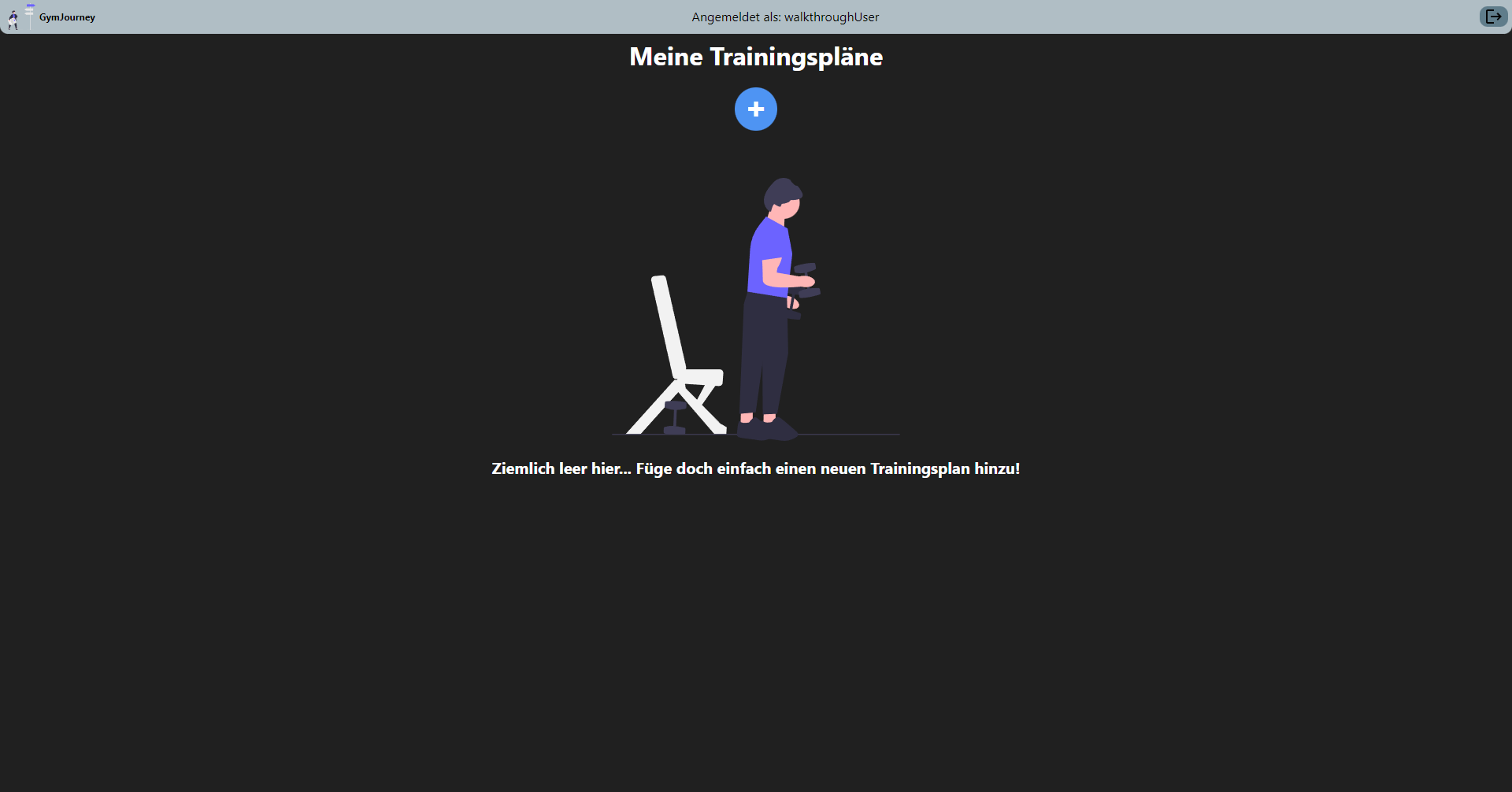
Automatisch generierte Beschreibung

**Abbildung 6:** Register-Seite

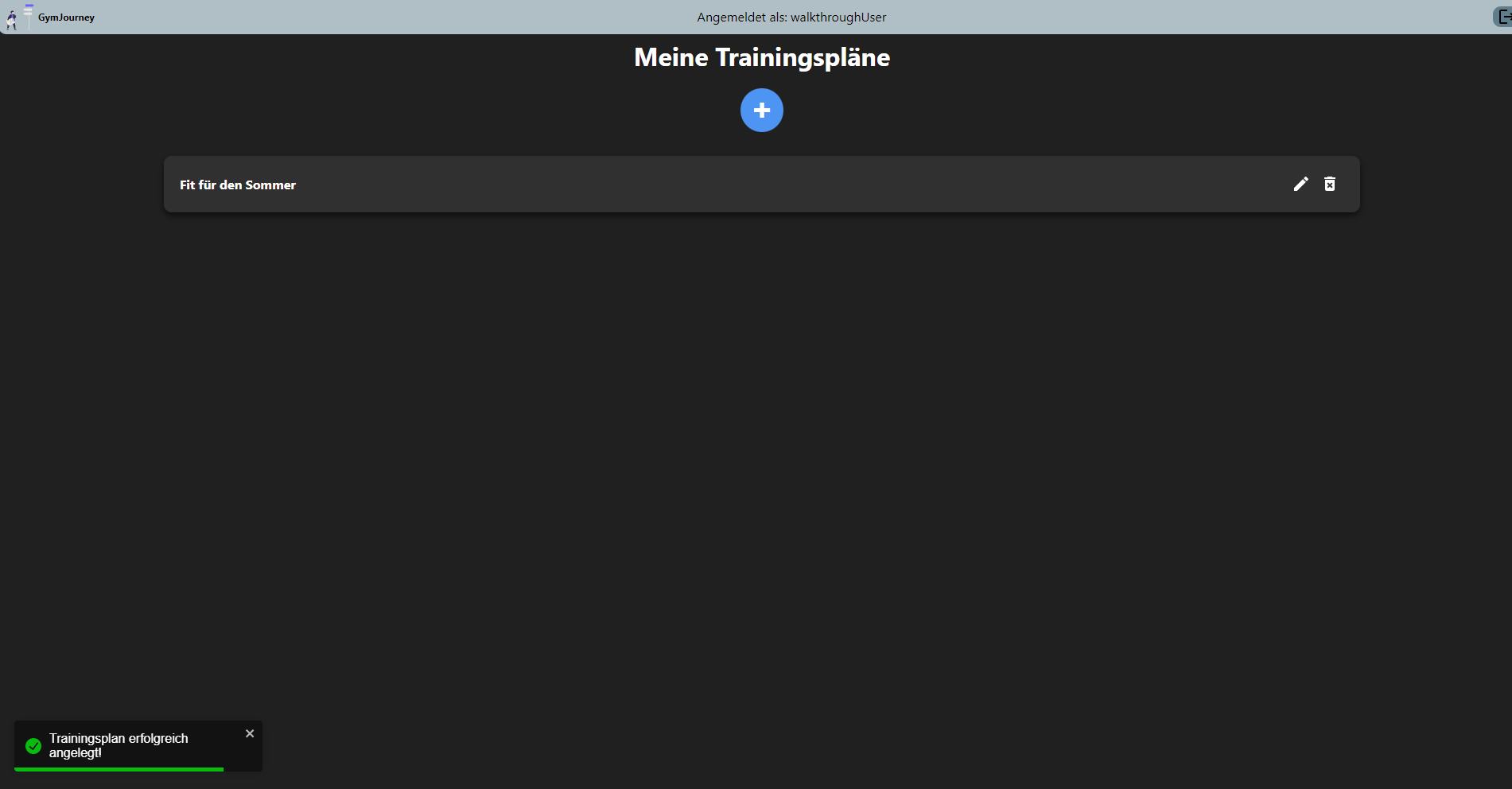
**Abbildung 5:** Login-Seite

Sowohl nach einem erfolgreichen Login als auch nach einer erfolgreichen Registrierung landet man auf der in Abbildung 7 dargestellten Home-Seite. Am oberen Bildschirmrand ist eine Navigationsleiste zu sehen, die das Logo der GymJourney Website darstellt, mit dem man per Klick wieder auf die Home-Seite zurückkehren kann sowie den Namen des angemeldeten Nutzers und einen Button, mit dem sich der Nutzer abmelden kann. Diese Navigationsleiste wird nur bei angemeldeten Nutzern angezeigt. Auf der Home-Seite werden die Trainingspläne abgebildet. Da der Nutzer aber noch keine Trainingspläne erstellt hat, werden ihm auch keine angezeigt. Um einen neuen Trainingsplan zu erstellen, klickt man auf den „+“ Button. Daraufhin öffnet sich ein Fenster, in dem man den Namen für den Trainingsplan vergeben kann, wie Abbildung 8 zeigt.

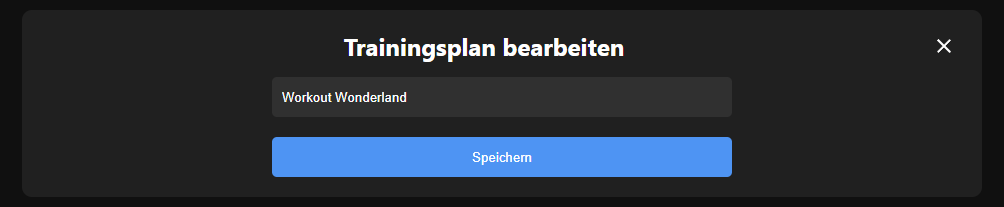
**Abbildung 8:** Trainingsplan erstellen

Klickt man auf den „Speichern“ Button des Fensters, so wird ein neuer Trainingsplan erstellt. Dieser Trainingsplan wird dann direkt auf der Home-Seite angezeigt zusammen mit einer Erfolgsmeldung über die Erstellung des Trainingsplans. Beides kann in Abbildung 9 gesehen werden.

**Abbildung 7:** Home-Seite

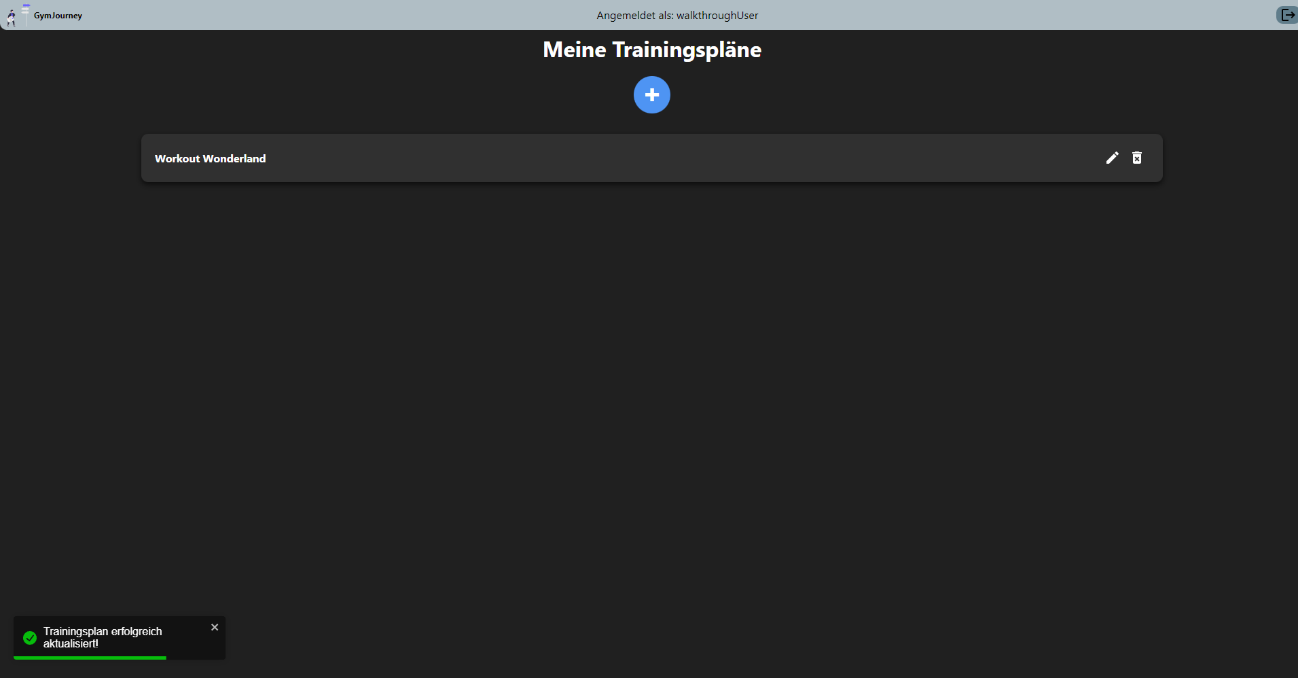
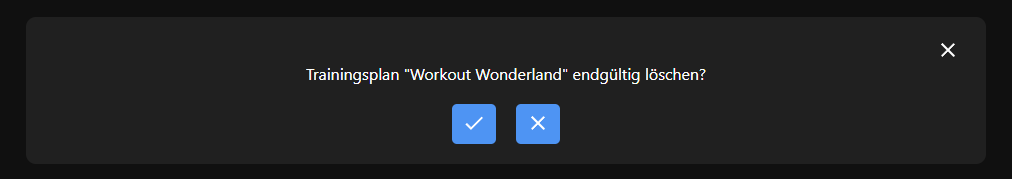
Klickt man auf das Stift-Icon im Container des erstellten Trainingsplans, so öffnet sich ein Fenster, bei dem man den Namen des Trainingsplans ändern kann (siehe Abbildung 10).

**Abbildung 9:** Der erstellte Trainingsplan



**Abbildung 10:** Trainingsplan bearbeiten

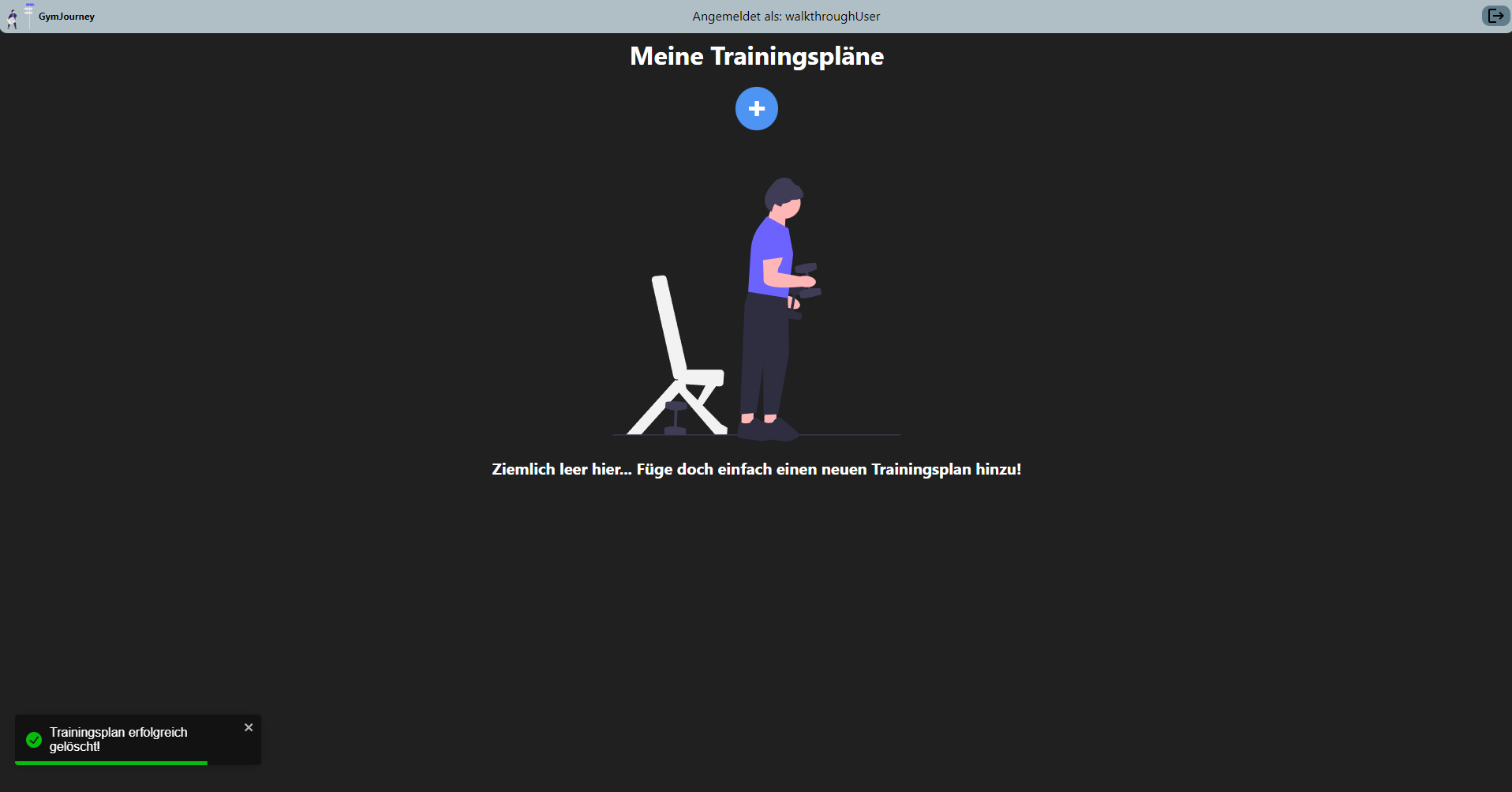
Betätigt man in diesem Fenster die Speichern Taste, so wird der neue Name des Trainingsplans übernommen und zusammen mit einer Erfolgsmeldung auf der Home-Seite dargestellt.

Klickt man nun auf das Mülleimer-Icon im Trainingsplan-Container, so kann der Trainingsplan gelöscht werden. Auch hierfür öffnet sich zunächst ein Fenster, das auf Abbildung 12 dargestellt wird.

**Abbildung 12:** Trainingsplan löschen

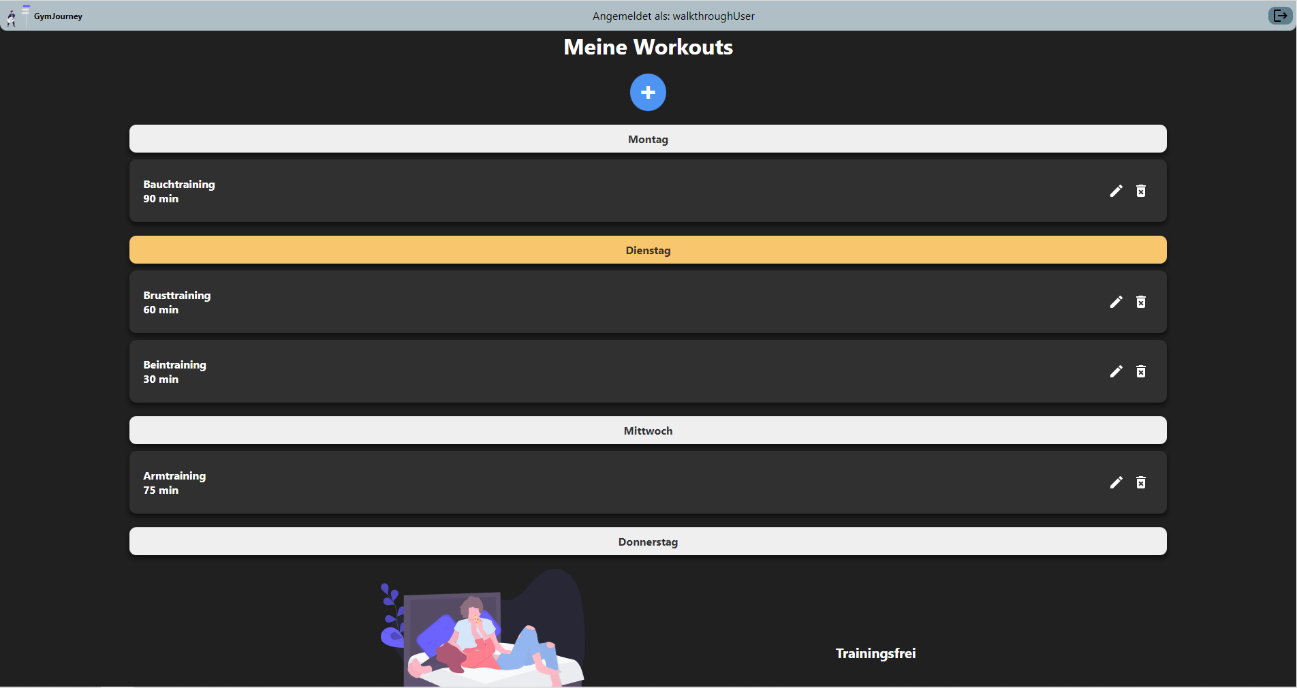
**Abbildung 11:** Der bearbeitete Trainingsplan

In diesem Fenster hat man nun die Möglichkeit die Aktion durch Klick auf den rechten Button abzubrechen, während der Trainingsplan durch klick auf den linken Button endgültig gelöscht wird und auch nicht wieder hergestellt werden kann. Wird ein Trainingsplan gelöscht, so werden auch alle zum Trainingsplan gehörigen Workouts und Übungen gelöscht. Führt man das Löschen aus, so erhält man auch hier wieder eine Erfolgsmeldung und sieht den gelöschten Trainingsplan nicht mehr auf der Home-Seite.

Klickt man auf einen Trainingsplan aus der Trainingsplan-Seite wie hier beispielsweise „Workout Wonderland“ aus Abbildung 11, so landet man auf der in Abbildung 14 dargestellten Workouts-Seite des jeweiligen Trainingsplans, bei der aufgelistet wird, welches Workout an welchem Wochentag geplant ist. Der aktuelle Tag wird hierbei farblich hervorgehoben. Da aktuell noch keine Workouts erstellt wurden, ist an keinem Tag ein Workout geplant.

**Abbildung 14:** Workouts-Seite

**Abbildung 13:** Erfolgreiche Löschung eines Trainingsplans

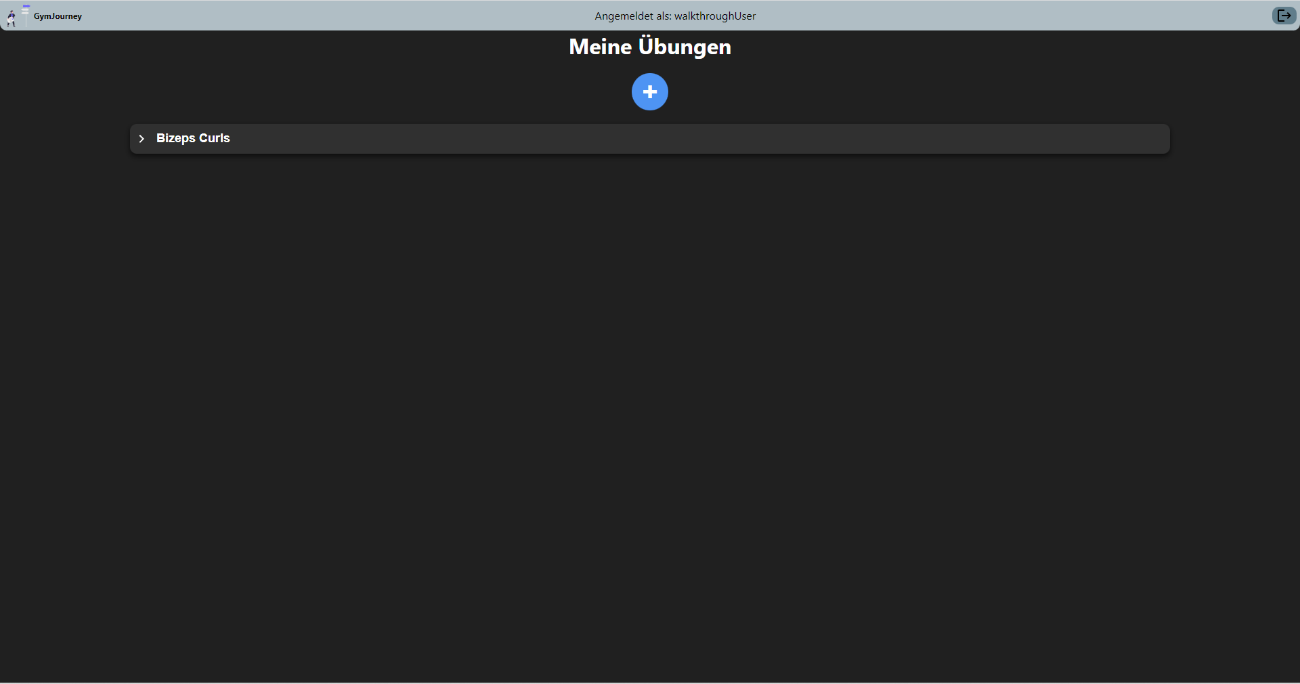
Hier kann durch einen Klick auf den „+“-Button nun ein neues Workout erstellt werden. Da sich hier ein ähnliches Fenster öffnet wird hier auf eine Abbildung verzichtet. Der einzige Unterschied ist lediglich, dass hier andere Angaben gemacht werden müssen. Es muss ein Wochentag aus einem Drop-Down-Menü ausgewählt werden und der Name des Workouts sowie die Dauer dieses Workouts in Minuten müssen angegeben werden. Danach kann auf Speichern geklickt werden und das Workout wird abgebildet. In Abbildung 15 wird gezeigt, wie die Workouts-Seite mit einigen gespeicherten Workouts aussieht.

**Abbildung 15:** Workouts-Seite mit gespeicherten Workouts

**Abbildung 15:** Workouts-Seite mit gespeicherten Workouts

Möchte man ein Workout bearbeiten, so klickt man auch hier auf das Stift-Icon und es öffnet sich auch hier ein PopUp wie in Abbildung 10, in dem man nun den Wochentag, den Namen und die Länge des Trainingsplans ändern kann. Für das Löschen eines Workouts betätigt man das Mülleimer-Icon und bestätigt in einem Ähnlichen PopUp wie in Abbildung 12 die Löschung des Workouts. Wird ein Workout gelöscht, so werden auch alle zum Workout gehörigen Übungen gelöscht.

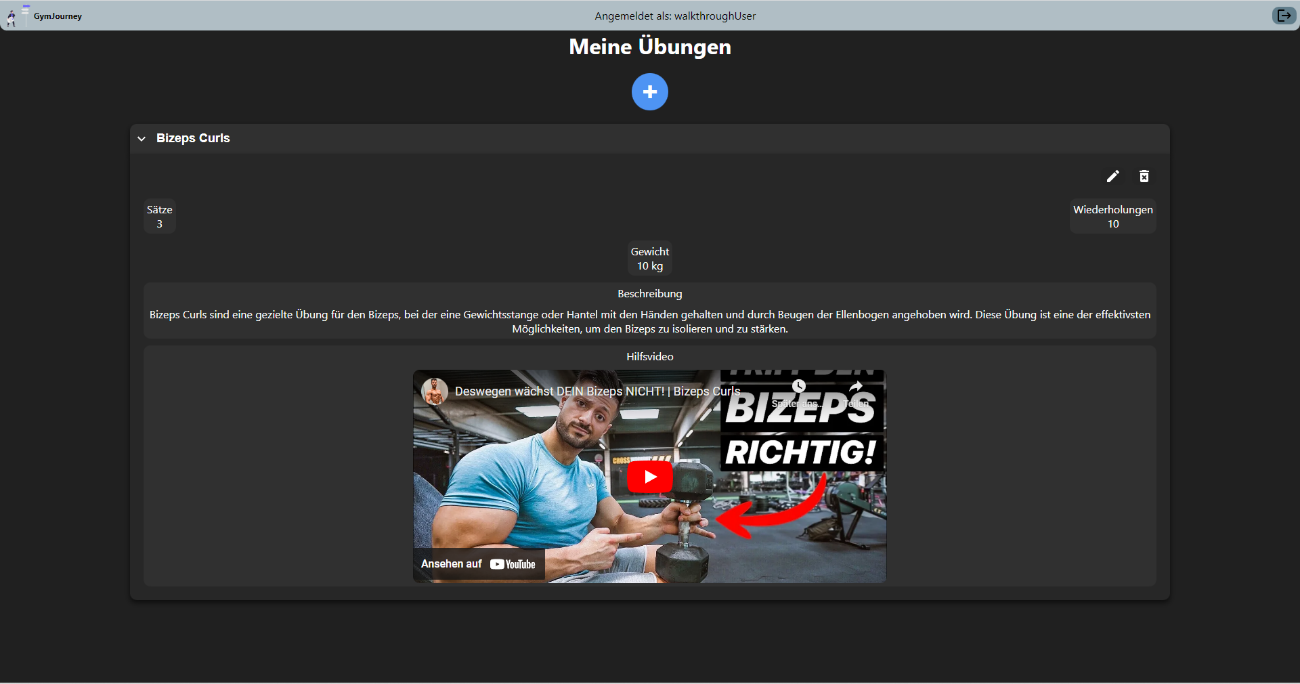
Für ein Workout können beliebig viele Übungen erstellt werden. Um Übungen für ein Workout zu erstellen, wählt man einen Workout aus der Liste wie in Abbildung 15 aus und landet auf der in Abbildung 16 dargestellten Übungen-Seite.

 Auch diese Seite ist zunächst leer, jedoch könne wie in Abbildung 8 durch Klick auf den „+“-Button durch ein PopUp neue Übungen erstellt werden. Hierfür **müssen** der Name der Übung, die Anzahl der Sätze sowie die Anzahl der Wiederholungen pro Satz angegeben werden, während das verwendete Gewicht für die Übung in kg, ein Link zu einem YouTube-Tutorial sowie die Beschreibung der Übung **freiwillig** sind. Abbildung 17 zeigt die Übungen-Seite nach dem Erstellen einer Übung. 

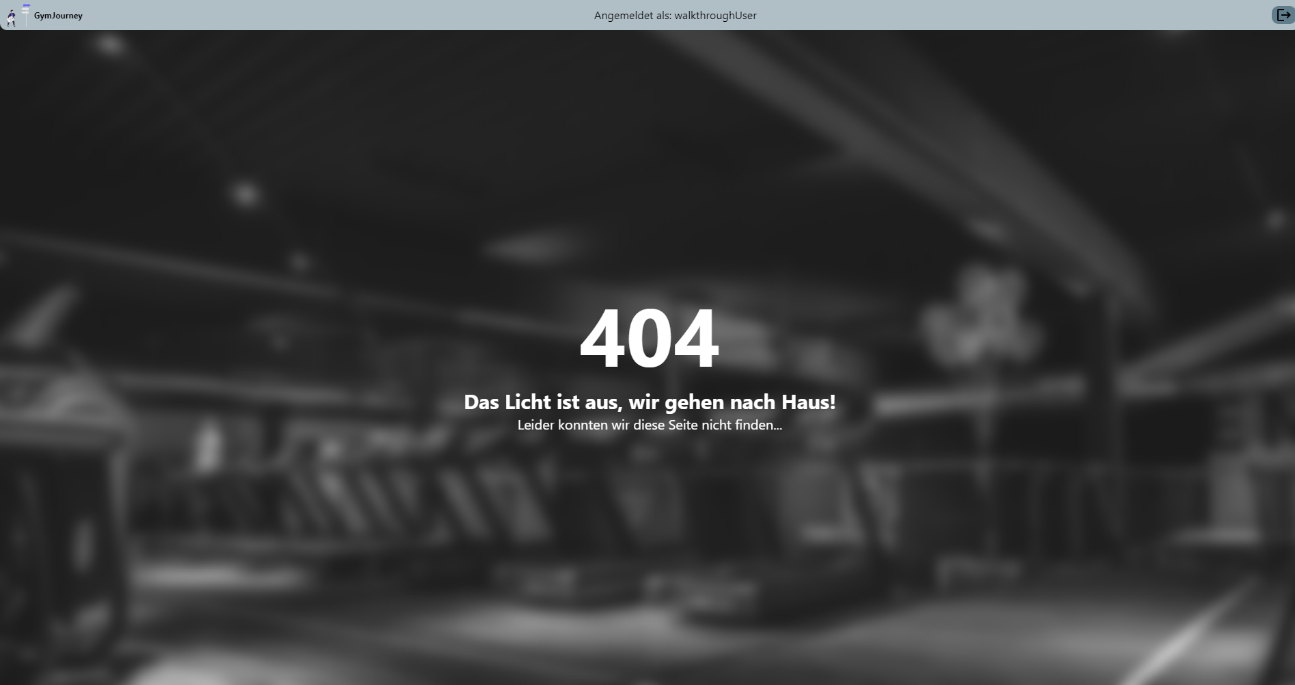
**Abbildung 17:** Übungen-Seite mit gespeicherter Übung

**Abbildung 16:** Übungen-Seite

Um weitere Informationen zu der Übung zu sehen, klickt man lediglich auf die Übung. Das Resultat davon zeigt Abbildung 18.

Hier erhält man nun auch die Möglichkeit die Übung durch Klick auf das Stift-Icon ähnlich wie in Abbildung 10 zu bearbeiten, in dem man nun jegliche Angabe, die man zur Übung gemacht hat, in einem PopUp bearbeiten kann. Durch einen Klick auf das Mülleimer-Icon ist es auch hier möglich nach Bestätigung eines PopUp wie in Abbildung 12 die Übung zu löschen.

**Abbildung 18:** Informationen zu einer Übung

Gelangt der Nutzer beispielsweise durch Eingabe einer URL auf einer nicht existierenden Seite oder versucht der Nutzer durch Änderung einer ID in der URL auf eine Seite eines anderen Nutzers zuzugreifen, so wird ihm folgende Fehlerseite angezeigt.

**Abbildung 19:** Fehlerseite

//TODO: ZU FEHLER BEI NICHT ERLAUBTEN ZUGRIFF EVTL. SCHREIBEN, DASS MAN SICH AN GITHUB ORIENTIERT HAT

# Fazit

# Literaturverzeichnis

[1] Kinsta. *What Is React.js? A Look at the Popular JavaScript Library.* URL: <https://kinsta.com/knowledgebase/what-is-react-js/> (besucht am 26.04.2023).

[2] MDN Web Docs. *Express/Node introduction*. URL: <https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs/Introduction> (besucht am 26.04.2023).

[3] Kinsta. *Was ist MySQL? Eine anfängerfreundliche Erklärung.* URL: <https://kinsta.com/de/wissensdatenbank/was-ist-mysql/> (besucht am 26.04.2023).

[4] Iconfinder. *Js, react js, logo icon.* URL: <https://www.iconfinder.com/icons/1174949/js_react_js_logo_react_react_native_icon> (besucht am 26.04.2023).

[5] Wikimedia Commons. *File:Expressjs.png.* URL: <https://commons.wikimedia.org/wiki/File:Expressjs.png> (besucht am 26.04.2023).

[6] Creazilla, Gil Barbara. *Mysql icon.* URL: <https://creazilla.com/nodes/3254121-mysql-icon> (besucht am 26.04.2023).

Interessante Dinge, die ich genutzt habe:

JWT-Token und Cookies zur Authorisierung.

<https://www.youtube.com/watch?v=1EuNnZEp2sQ&list=PLrNy47mmRpmPj5_W4OdjvrBbP1LCEVgSF&index=99&t=126s&ab_channel=LamaDev>

Minute 34:15

* Dadurch muss man die User-Id nicht überall herausfinden/mitgeben, sondern kann den Cookie nutzen

Ich nutze stand jetzt localStorage um den Nutzer dauerhaft zu speichern (I guess)

React Query (siehe verlinktes Video)

Bei Home nutze ich Home.css statt [Home.module.css](http://Home.module.css) da sonst das ändern der Styles des Popups nicht möglich ist

<https://react-popup.elazizi.com/css-styling>

forwardRef verwendet im Button weil es sonst in popup-js zu folgendem Fehler kam:

Check the render method of `ForwardRef`. at Button (http://localhost:3000/static/js/bundle.js:359:5) at http://localhost:3000/static/js/bundle.js:59182:27 at div at div at Home (http://localhost:3000/static/js/bundle.js:1086:68) at PrivateRoute (http://localhost:3000/static/js/bundle.js:279:5) at RenderedRoute (http://localhost:3000/static/js/bundle.js:53437:5) at Routes (http://localhost:3000/static/js/bundle.js:53927:5) at QueryClientProvider (http://localhost:3000/static/js/bundle.js:50471:21) at App at Router (http://localhost:3000/static/js/bundle.js:53865:15) at BrowserRouter (http://localhost:3000/static/js/bundle.js:52054:5) at AuthContextProvider (http://localhost:3000/static/js/bundle.js:868:5)