Prototyping Projektdokumentation

Name: Loïc Kaufmann

E-Mail: kaufmloi@students.zhaw.ch

URL der deployten Anwendung: https://habittrackerln2.netlify.app

# Einleitung

App-Name: DailyHabit

DailyHabbit ist eine Webanwendung, welche zur Unterstützung beim Aufbau und Verfolgen von gesunden Gewohnheiten dienen sollen. Die App erlaubt es den Benutzern, eigene Habits zu definieren, diese zu tracken und auszuwerten. Die Grundidee der App ist es langfristig Verhaltensweisen durch kontinuierliche Wiederholung zu etablieren. Diese werden anhand Tagesfortschritten, Streaks und einer Monatsübersicht visualisiert.

Die Grundfunktionen der App sind:

* Erstellen, Anzeigen und Löschen von Habits
* Tracken der täglichen und wöchentlichen Habit-Erfüllungen
* Anzeige von Fortschritt
* Streak-Logik
* Kalender
* Kategorisierung von Habits

# Datenmodell

*[ER-Diagramm (mit Erläuterungen, sofern nicht selbsterklärend)]*

# Beschreibung der Anwendung

*[Beschreibung der einzelnen Pages und Funktionen inklusive Screenshots. Die Funktionalität und die Workflows müssen anhand der Screenshots und Textbeschreibungen nachvollziehbar sein. Der Code muss nicht beschrieben werden.]*

*Beispiel:*

## Filmübersicht

Route: /movies

# 

Auf dieser Seite werden alle Filme aufgelistet. Für die Darstellung der einzelnen Filme wird die Komponente MovieCard.svelte verwendet. Jeder Film kann zur Watchlist hinzugefügt oder von dieser entfernt werden. Per Checkbox «Show only movies on watchlist» können die Filme ausgeblendet werden, die nicht auf der Watchlist stehen. Über den Button «Add New Movie» gelangt man zur Seite /movies/create.

Dateien:

* lib/components/MovieCard.svelte
* routes/movies/+page.svelte
* routes/movies/+page.server.js

# Erweiterungen

## Streak-Logik:

Auf der Statistiksteite /stats und der Detailseite /habits/[id] werden für jedes Habit aufeinanderfolgende erfüllte Tage einer Habit als «Streaks» gezählt und angezeigt werden. Die Berechnung der «Streak-Logik» erfolgt über die Funktion getStreakForHabit().

Dateien:

* lib/db/entries.js
* routes/stats/+page.server.js
* routes/habits/[id]/+page.server.js

## Kalenderansicht:

Auf der Statistikseite /stats wird ein Monatskalender angezeigt, der den täglichen Fortschritt aller Habits pro Tag zusammenfasst. Die Anzeige des Kalenders basiert auf den gespeicherten entries.

Dateien:

* src/lib/components/CalendarView.svelte
* routes/stats/+page.svelte
* routes/stats/+page.server.js

## Kategorie-System mit eigener Collection:

Habits sind über ein Dropdown einer Kategorie zugeordnet. Kategorien sind als eigene Collection (categories) in der Datenbank gespeichert. Die Verwaltung erfolgt über die Seite /categories.

Dateien:

* lib/db/categories.js
* routes/categories/+page.svelte
* routes/habits/new/+page.svelte
* routes/habits/new/+page.server.js

## Responsive-Layout:

Auf der gesamten Webanwendung wird Bootstrap 5 verwendet und ist dadurch auf Mobilgeräten auch vollständig responsive. Die Navigation ist als einklappbare Mobile-Navbar implementiert.

Dateien:

* src/app.hmtl
* routes/+layout.svelte
* Globale Verwendung von Bootstrap-Klassen (container, row, col, btn usw.)

## Bearbeitungsfunktion für Habits:

Auf der Detailseite eines Habits (/habits/[id]) befindet sich ein Button zum Bearbeiten. Dieser führt zur Seite /habits/[id]/edit, wo die bestehenden Werte vorausgefüllt sind und geändert werden können.

Dateien:

* routes/habits/[id]/+page.svelte
* routes/habits/[id]/edit/+page.svelte
* routes/habits/[id]/edit/+page.server.js