

Prototyping Projektdokumentation

Name: Elza Miftari

E-Mail: miftaelz@students.zhaw.ch

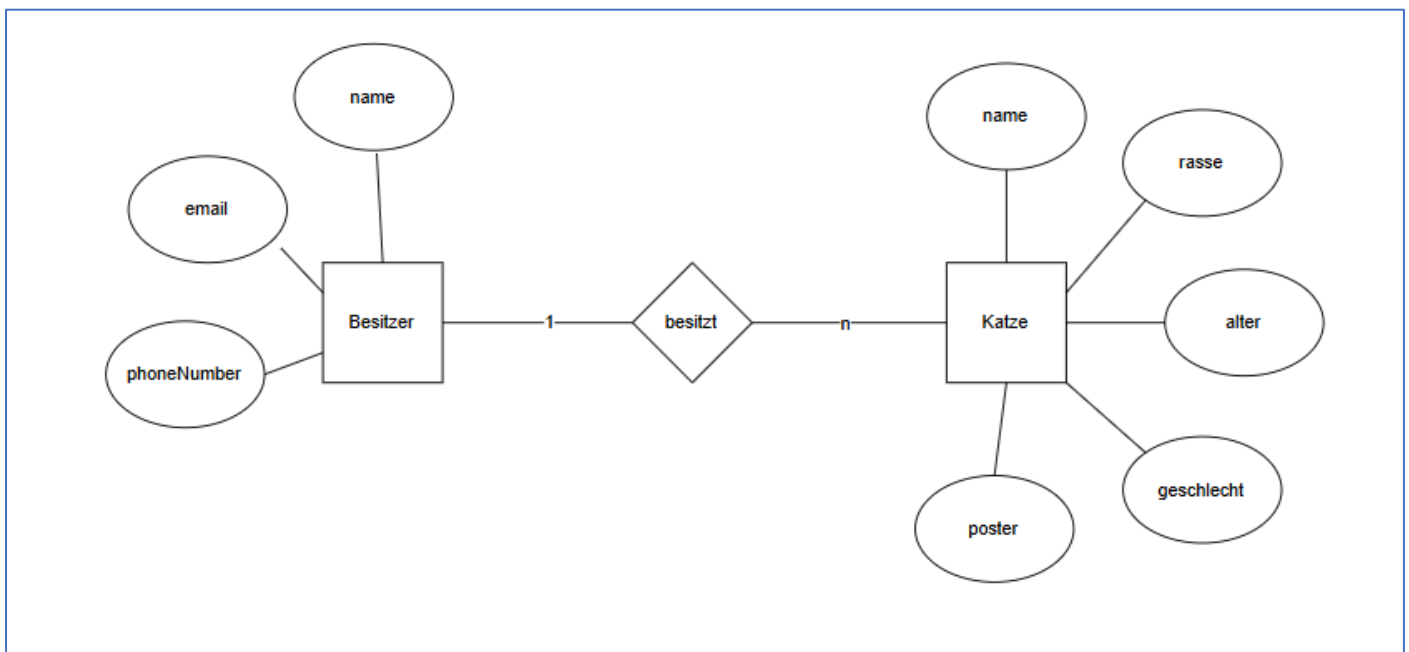
URL der deployten Anwendung: <https://katzenausstellung.netlify.app/>

1. Einleitung

Die Webanwendung «**Katzen-Community**» ermöglicht das Verwalten von Katzen und ihren Besitzerinnen. Ziel der Anwendung ist es, Katzen in einer Galerie darzustellen, neue Tiere hinzuzufügen, Details anzuzeigen sowie die zugehörigen Besitzerinnen zu verwalten. Zusätzlich können Besitzerprofile bearbeitet oder gelöscht werden.

Die Anwendung verwendet SvelteKit im Runes-Mode, MongoDB zur Datenspeicherung sowie HTML, CSS und Bootstrap zur Gestaltung. Das Projekt hat zwei Entitätstypen: **Katze** und **Besitzer**.

2. Datenmodell



3. Beschreibung der Anwendung

3.1. Startseite (Home)

Route: /routes

Willkommen im digitalen Katzenuniversum

Hier regieren Schnurrhaare, Pfoten und Persönlichkeit.

Egal ob flauschig, frech, königlich oder chaotisch - zeig der Welt deine Katze und dich als stolzen Besitzer oder stolze Besitzerin!

Lade ein Bild hoch, gib ihr einen Namen, erzähl was über sie -

und lass sie auf unserer digitalen Bühne glänzen.

Kein Wettbewerb, kein Stress. Nur Katzenliebe pur. 🐾

Was du tun kannst:

- Durch andere Katzen stöbern
- Deine eigene Katze hinzufügen
- Dich als Besitzer*in registrieren
- Vielleicht sogar bald liken, kommentieren oder abstimmen?

Online. Flauschig. Kostenlos. Willkommen in der Katzengalaxie. 🐱 ✨

Klicke hier, um deine Katze hinzuzufügen und Teil der Community zu werden!

Katze hinzufügen

Auf der Startseite werden die Besucher direkt angesprochen und in die Welt der Anwendung eingeführt. Die Navigation oben bietet Zugriff auf die beiden Hauptbereiche «Katzen» und «Besitzer».

Der zentral platzierte Bereich beschreibt die Idee der Plattform: eine liebevoll gestaltete Community rund um Katzen, ganz ohne Wettbewerbsdruck – mit dem Fokus auf Teilen, Entdecken und Mitmachen.

Durch den Button «Katze hinzufügen» auf der Startseite kann eine neue Katze direkt erfasst werden, ohne zuerst über die Navigation zur Seite «Katzen» zu wechseln.

Datei:

- routes/+page.svelte

3.2. «Katzen» Übersicht

Route: /katzen

Entdecke die Katzencommunity

Suche nach einer Katze...

Hier präsentieren stolze Katzenbesitzer*innen ihre ganz besonderen Lieblingsstars. Von gemütlichen Couch-Katzen bis hin zu selbstbewussten Showkatzen - klick dich durch unsere Sammlung und entdecke die unterschiedlichsten Persönlichkeiten auf vier Pfoten!

Katze hinzufügen



Luna
Rasse: Siamkatze
Alter: 2



Simba
Rasse: Maine Coon
Alter: 4



Leo
Rasse: Bengalkatze
Alter: 1



Cleo
Rasse: Norwegische Waldkatze
Alter: 7

Auf der Seite «Katzen» wird eine Galerie aller in der Datenbank erfassten Katzen angezeigt. Nutzer*innen können bequem durch die Sammlung blättern oder über die integrierte Suchleiste gezielt nach bestimmten Katzen suchen.

Ein auffälliger Button ermöglicht es, direkt eine neue Katze zu erfassen.

Beim Hover-Effekt wird die ausgewählte Card hervorgehoben, was die Benutzerführung verbessert. Zusätzlich sind sowohl das Katzenbild als auch der Name klickbar. Beim Überfahren mit der Maus wird der Name blau und unterstrichen dargestellt. Ein Klick führt auf die Detailseite der jeweiligen Katze.

Dateien:

- routes/katzen/+page.svelte
- routes/katzen/+page.server.js
- lib/components/KatzenCard.svelte
- lib/db.js

3.3. Katze hinzufügen Button

Route: /katzen/create

Füge deine Katze hinzu

Name

Rasse

Alter

Geschlecht

Speichern

Zurück zur Übersicht

Sobald man auf «Katze hinzufügen» klickt, erscheint ein Formular, in dem der Nutzer die Angaben zur neuen Katze einträgt und sie anschliessend speichert.

Füge deine Katze hinzu

Name

Milo

Rasse

BKH

Alter

1

Geschlecht

Männlich

Speichern

Nach dem Klick auf «Speichern» wird eine grüne Meldung angezeigt, die den erfolgreichen Eintrag bestätigt.

Füge deine Katze hinzu

Name

Rasse

Alter

Geschlecht

Speichern

Katze wurde erfolgreich hinzugefügt

Zurück zur Übersicht

Mit dem Button «Zurück zur Übersicht» gelangt man dann wieder zur Übersicht der Katzen.

Dateien:

- routes/katzen/create/+page.svelte
- routes/katzen/create/+page.server.js
- lib/db.js

3.4. Detailseite einer Katze

Route: /katzen/[katzen_id]

Kiro



Rasse: Britisch Kurzhaar

Alter: 1

Geschlecht: Männlich

Katze entfernen

Katzendaten bearbeiten

Zurück zur Übersicht

Sobald man auf den Namen oder das Bild einer Katze klickt, gelangt man zur Detailseite. Dort kann man die Katze entweder löschen, ihre Daten bearbeiten oder zur Übersichtsseite zurückkehren. Wenn man auf das Button «Katze entfernen» klickt, wird die Card gelöscht und geht automatisch wieder in die Übersichtsseite zurück.

Dateien:

- routes/katzen/[katzen_id]/+page.svelte
- routes/katzen/[katzen_id]/+page.server.js
- lib/db.js

3.5. Katzendaten bearbeiten

Route: /katzen/[katzen_id]/edit

[Home](#) [Katzen](#) [Besitzer](#)

Katze bearbeiten

Name

Rasse

Alter

Geschlecht

Bild-URL

Änderungen speichern

Abbrechen

Wenn man auf den Button «Katzendaten bearbeiten» klickt, erhält man die Möglichkeit, die Informationen der Katze anzupassen. Mit «Änderungen speichern» werden die Eingaben übernommen und man wird zurück zur Detailseite weitergeleitet. Mit «Abbrechen» gelangt man ebenfalls zurück, jedoch ohne dass Änderungen gespeichert werden.

Dateien:

- routes/katzen/[katzen_id]/edit/+page.svelte
- routes/katzen/[katzen_id]/edit/+page.server.js
- lib/db.js

3.6. «Besitzer» Übersicht

Route: /besitzer

Unsere stolzen Besitzer*innen

Hier findest du eine Übersicht aller registrierten Katzenbesitzer*innen - von fürsorglichen Alltagsbegleitern bis hin zu passionierten Ausstellungsprofis, die ihre Vierbeiner mit Stolz präsentieren.

Besitzer hinzufügen

Ashleigh Nassey Email: anassey0@qq.com Telefon: 678-498-2518	Braden Stanfield Email: bstanfield1@taobao.com Telefon: 782-356-2644	Abbe Fordham Email: afordham2@dailymail.co.uk Telefon: 130-817-6125	Putnem Mighele Email: pmighele3@dailymail.co.uk Telefon: 317-795-2709
Kendricks Zuann Email: kzuann4@oracle.com Telefon: 605-827-1592	Dominica Grinter Email: dgrinter5@ow.ly Telefon: 315-518-3602	Zara Hardern Email: zhardern6@adobe.com Telefon: 332-297-1937	Frannie Risen Email: frisen7@who.int Telefon: 619-796-7152
Kenn Ledingham Email: kledingham8@istockphoto.com Telefon: 360-108-4550	Halsy McLaverty Email: hmclaverty9@telegraph.co.uk Telefon: 903-261-0236		

Daten generiert mit ChatGPT

Auf der Seite «Besitzer» wird eine Übersicht aller registrierten Katzenbesitzer*innen angezeigt. Beim Überfahren der Cards mit der Maus werden diese optisch hervorgehoben, um die Benutzerführung zu unterstützen.

Angezeigt werden der Name, die E-Mail-Adresse sowie die Telefonnummer der jeweiligen Person. Der Name ist klickbar, beim Darüberfahren wird er blau markiert und führt per Klick zur Detailseite des Besitzers.

Über den blauen Button «Besitzer hinzufügen» kann eine neue Person zur Datenbank hinzugefügt werden.

Dateien:

- routes/besitzer/page.svelte
- routes/besitzer/page.server.js
- lib/components/BesitzerCard.svelte
- lib/besitzerdb.js

3.7. Besitzer hinzufügen

Route: /besitzer/create

Besitzer hinzufügen

Name

emi

E-Mail

test@ch.com

Telefonnummer

5555555555

Speichern

Zurück zur Übersicht

Sobald man den Button «Besitzer hinzufügen» klickt, kann ein neuer Besitzer erfasst werden.

Besitzer hinzufügen

Name

E-Mail

Telefonnummer

Speichern

Besitzer wurde erfolgreich hinzugefügt

Zurück zur Übersicht

Nach dem Speichern kommt eine Meldung in grün, dass der Besitzer erfolgreich hinzugefügt wurde. Mit dem Button «Zurück zur Übersicht» gelangt man wieder auf die Übersichtsliste der Besitzer.

Dateien:

- routes/besitzer/create/page.svelte
- routes/besitzer/create/page.server.js
- lib/besitzerdb.js

3.8. Detailseite Besitzer

Route: /besitzer/[besitzer_id]

Besitzer-Details

emi

Email: test@ch.com

Telefon: 555555555555

Besitzer*in entfernen

Besitzerdaten bearbeiten

Zurück zur Übersicht

Auch hier kann man den Besitzer durch die Buttons entweder löschen, die Daten bearbeiten oder zur Übersicht zurückkehren. Beim Löschen wird der Eintrag aus der Datenbank entfernt und man wird automatisch zur Übersichtsliste weitergeleitet.

Dateien:

- routes/besitzer/[besitzer_id]/page.svelte
- routes/besitzer/[besitzer_id]/page.server.js
- lib/besitzerdb.js

3.9. Besitzerdaten bearbeiten

Route: /besitzer/[besitzer_id]/edit

Besitzer bearbeiten

Name

emila

E-Mail

test@ch.com

Telefonnummer

555555555555

Änderungen speichern

Abbrechen

Nachdem man beispielsweise den Namen angepasst hat, kann man auf den Button «Änderung speichern» klicken, um die Änderungen zu übernehmen. Mit dem Button «Abbrechen» gelangt man zurück zur Detailseite, ohne dass die vorgenommenen Änderungen gespeichert werden.

Besitzer-Details

emila

Email: test@ch.com

Telefon: 555555555555

Besitzer*in entfernen

Besitzerdaten bearbeiten

Zurück zur Übersicht

Besitzer Daten wurden nun angepasst.

Dateien:

- routes/besitzer/[besitzer_id]/edit/page.svelte
- routes/besitzer/[besitzer_id]/edit/page.server.js
- lib/besitzerdb.js

4. Erweiterungen

Erweiterungen: Bearbeitungsfunktion, Suchfunktion, Hover-Effekt

4.1. Bearbeitungsfunktion für Besitzer

Besitzer*innen können in der Anwendung nicht nur erfasst und angezeigt, sondern auch nachträglich bearbeitet werden. Um diese Funktion zu ermöglichen, wurde unter routes/besitzer/[besitzer_id]/ ein zusätzlicher Ordner namens edit erstellt. Dieser enthält eine eigene +page.svelte und +page.server.js, welche das Bearbeitungsformular und die zugehörige Serverlogik bereitstellen.

Die dafür benötigten Daten werden mithilfe der Funktion getBesitzerById() aus der Datei lib/besitzerdb.js geladen. Nach dem Absenden des Formulars wird der Eintrag mit updateBesitzer() aktualisiert. Bei erfolgreichem Speichern wird der Nutzer automatisch zurück zur Detailseite des Besitzers weitergeleitet.

Dateien:

- routes/besitzer/[besitzer_id]/edit/+page.svelte
- routes/besitzer/[besitzer_id]/edit/+page.server.js
- lib/besitzerdb.js

4.2. Suchfunktion in der Katzenübersicht

Die Suchfunktion wurde in der Datei +page.svelte der Katzenliste implementiert. Ein Eingabefeld ist über bind:value={search} mit einer Variablen verknüpft. Mithilfe einer reaktiven \$:-Anweisung wird daraus eine gefilterte Liste (filteredKatzen) erzeugt, die im Template mit {#each} dargestellt wird.

Die Suchfunktion ist case-insensitive und funktioniert vollständig im Frontend, basierend auf den Daten, die aus +page.server.js über die load()-Funktion bereitgestellt werden.

Dateien:

- routes/katzen/+page.svelte
- routes/katzen/+page.server.js
- lib/db.js

4.3. CSS-Hover-Effekt in der KatzenCard

Das Hervorheben der Katzen-Cards beim Darüberfahren wurde mithilfe von CSS direkt in der Datei KatzenCard.svelte umgesetzt. Beim Hover wird die Card durch eine sanfte Animation optisch hervorgehoben:

- Vergrößerung per `scale(1.02)`
- Leichte Verschiebung nach oben mit `translateY(-4px)`
- Schatteneffekt durch `box-shadow`
- Übergangseffekt via `transition` für flüssige Bewegung

Dadurch wird visuell klar, welche Karte gerade fokussiert ist – was die Benutzerfreundlichkeit erhöht.

Datei:

- `lib/components/KatzenCard.svelte`