

Prototyping Projektdokumentation

Name: Ioannis Organitzidis

E-Mail: organioa@students.zhaw.ch

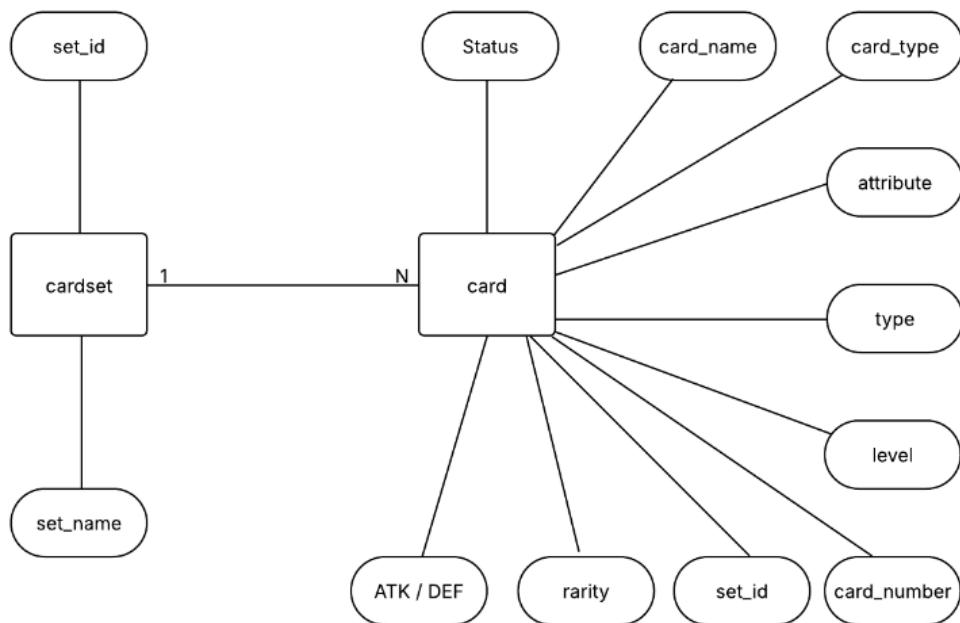
URL der deployten Anwendung: <https://yugiohdb.netlify.app/>

1. Einleitung

DuelDeck ist eine Webanwendung zur Verwaltung von Yu-Gi-Oh!-Sammelkarten. Nutzer können darin durch eine Kartendatenbank stöbern, einzelne Karten im Detail betrachten, neue Karten hinzufügen und ein persönliches Deck („My Deck“) zusammenstellen. Zusätzlich gibt es eine Übersicht über alle verfügbaren Kartensets, die ebenfalls im Detail durchsuchbar sind.

Die Anwendung wurde mit SvelteKit im Runes-Mode umgesetzt und verwendet MongoDB zur Datenverwaltung. Ziel war es, eine nutzerfreundliche, moderne App zu erstellen, mit der man bequem eine eigene Yu-Gi-Oh!-Kartensammlung online organisieren kann. Zu den Grundfunktionen zählen eine suchbare Kartenübersicht, Detailansichten, ein eigenes Deck, sowie das Erstellen und Löschen von Karten. Die Benutzeroberfläche wurde im Stil eines digitalen Sammelalbums gestaltet.

2. Datenmodell

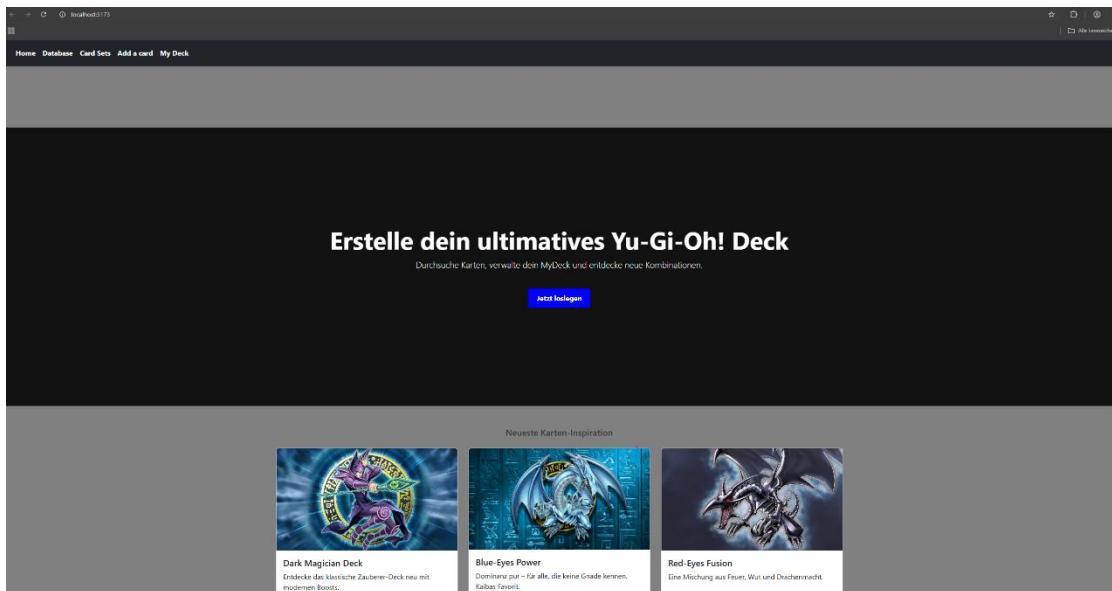


*Das Attribut **set_id** ist ein Fremdschlüssel (Foreign Key), der auf das zugehörige Kartenset verweist.

3. Beschreibung der Anwendung

3.1. Homepage

Route: /routes (Startseite)



Die Startseite von DuelDeck begrüßt die Nutzerinnen und Nutzer mit einem grosszügigen Hero-Bereich, der zur Erstellung eines eigenen Yu-Gi-Oh!-Decks einlädt. Ein zentral platziertter Button mit dem Text „Jetzt loslegen“ führt direkt zur Kartenübersicht. Unterhalb des Hero-Bereichs werden drei Beispielkarten mit Bild, Titel und kurzer Beschreibung dargestellt – sie dienen als visuelle Inspiration und verdeutlichen die Funktionalität der Anwendung. Die Navigationsleiste am oberen Bildschirmrand ist durchgängig verfügbar und erlaubt den Zugriff auf alle Hauptseiten der App:

Home, Database, Card Sets, Add a Card und My Deck.

Dateien:

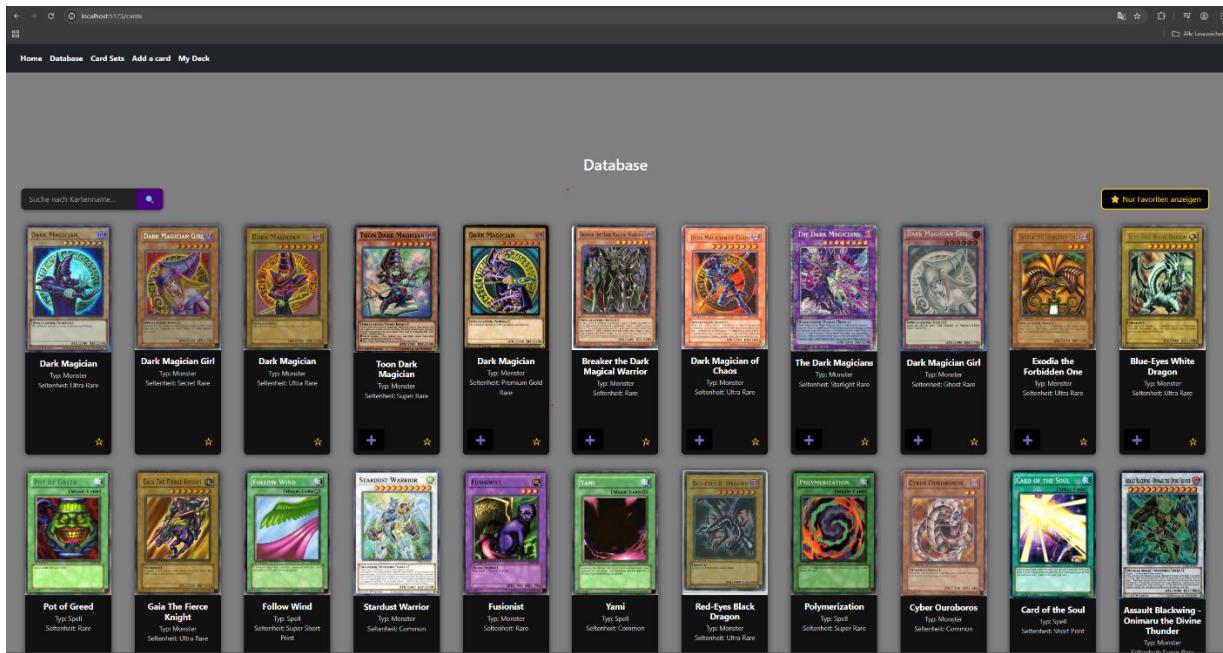
static/images

routes/+page.svelte

routes/+layout.svelte

3.2 Database

Route: /cards



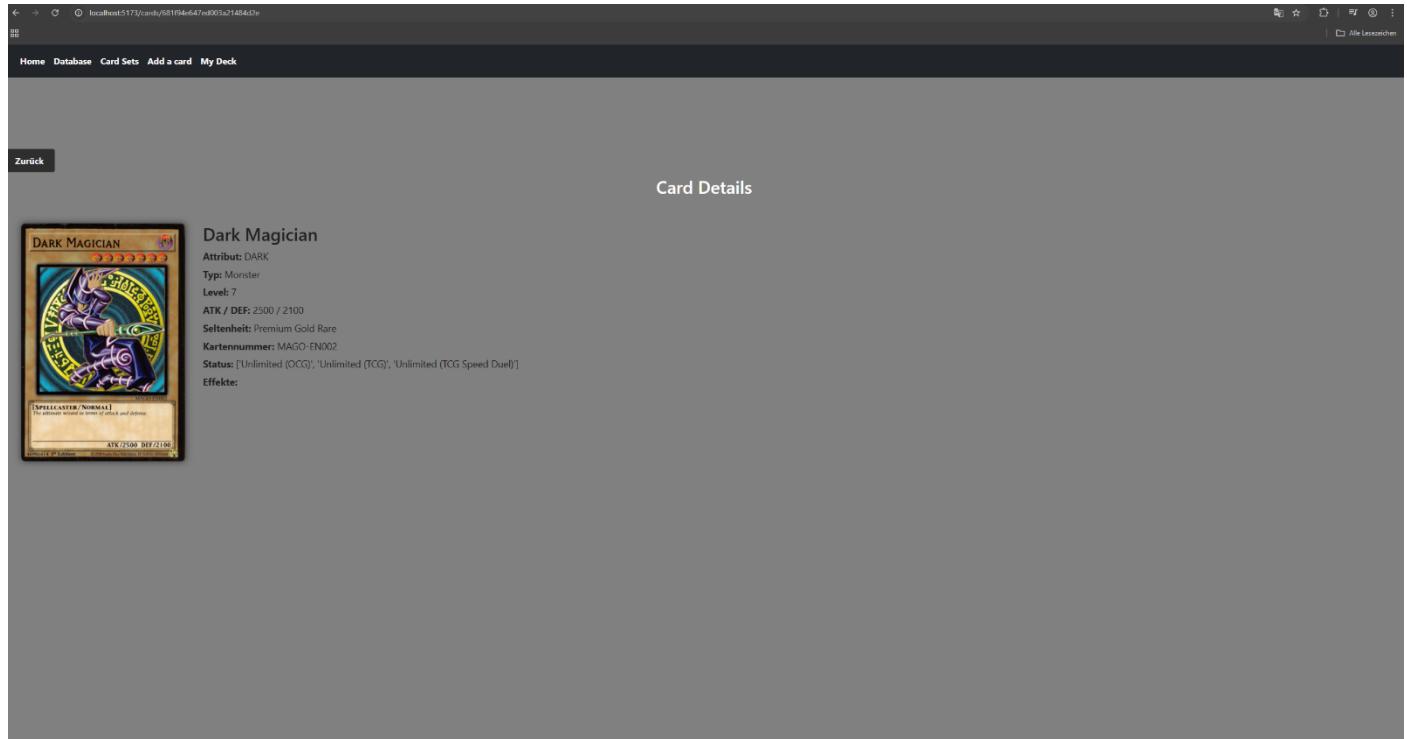
Auf dieser Seite werden alle Karten aus der Datenbank angezeigt. Nutzerinnen und Nutzer können durch die gesamte Sammlung scrollen oder mithilfe eines Suchfeldes gezielt nach bestimmten Karten suchen. Die Suche erfolgt serverseitig per Regex über den Kartennamen. Jede Karte wird mit **Bild**, **Name**, **Typ** und **Seltenheit** dargestellt. Über das **Plus-Symbol** (+) kann eine Karte direkt zum persönlichen Deck („**My Deck**“) hinzugefügt werden. Zusätzlich besteht die Möglichkeit, Karten als Favoriten zu markieren – über ein **Stern-Symbol** (★). Ist der Favoriten-Filter aktiviert, werden nur die Karten angezeigt, die zuvor als Favorit markiert wurden.

Dateien:

- lib/components/CardItem.svelte
- routes/cards/+page.svelte
- routes/cards/+page.server.js
- lib/db.js

3.3 Database (Detailseite)

Route: /cards/[id]



Auf dieser Seite werden die Details einer einzelnen Yu-Gi-Oh!-Karte angezeigt. Die Daten werden aus der MongoDB geladen und über die load()-Funktion an die Komponente CardDetail.svelte übergeben.

Die Detailansicht umfasst folgende Informationen:

- Bild der Karte
- Name, Attribut, Typ, Level
- ATK/DEF-Werte
- Seltenheit und Kartennummer
- Status und Effekttext

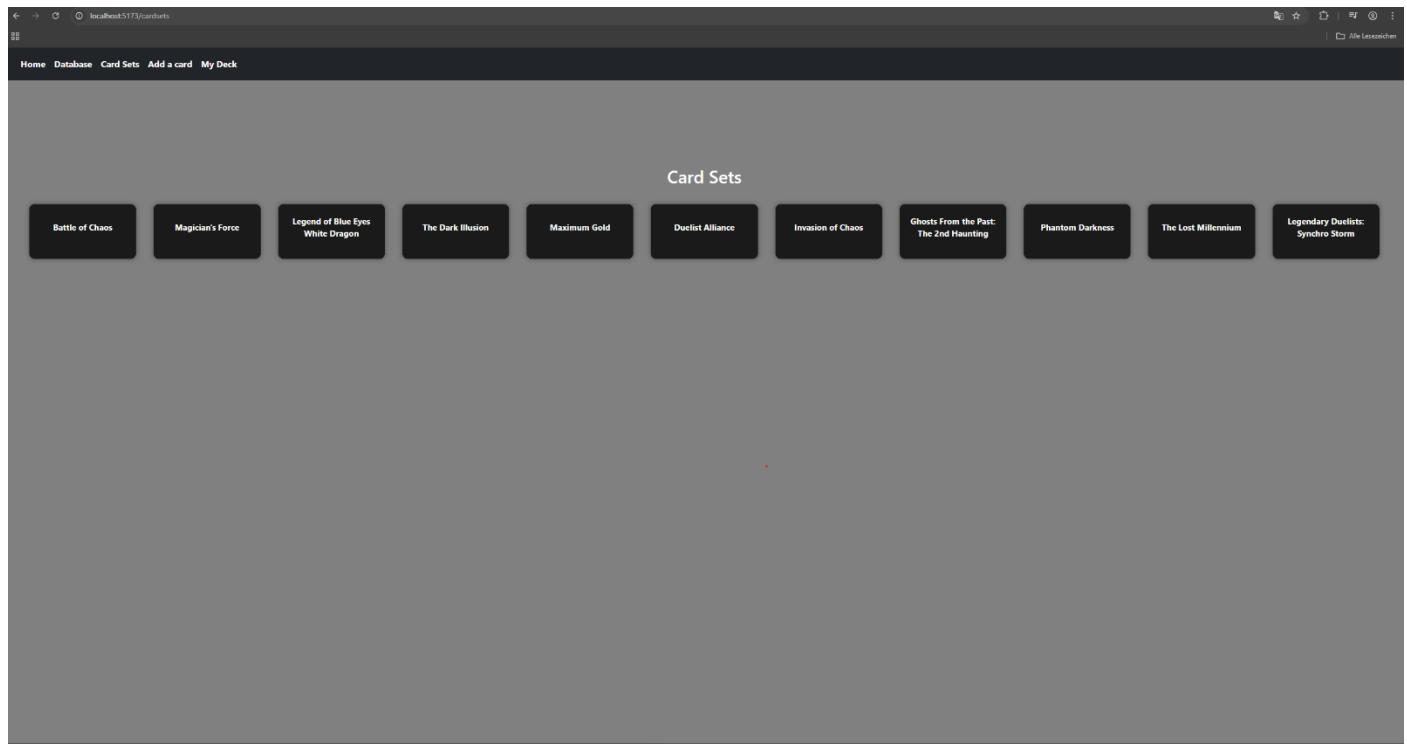
Ein „Zurück“-Button oberhalb der Karte führt zurück zur Kartenübersicht. Die Darstellung erfolgt in einem aufgeräumten Layout mit zentral platziertem Überschrift „**Card Details**“ und stilisierten Designelementen im dunklen Theme.

Dateien:

- lib/components/CardDetail.svelte
- routes/cards/[id]/+page.svelte
- routes/cards/[id]/+page.server.js
- lib/db.js

3.4 Card Sets

Route: /cardsets



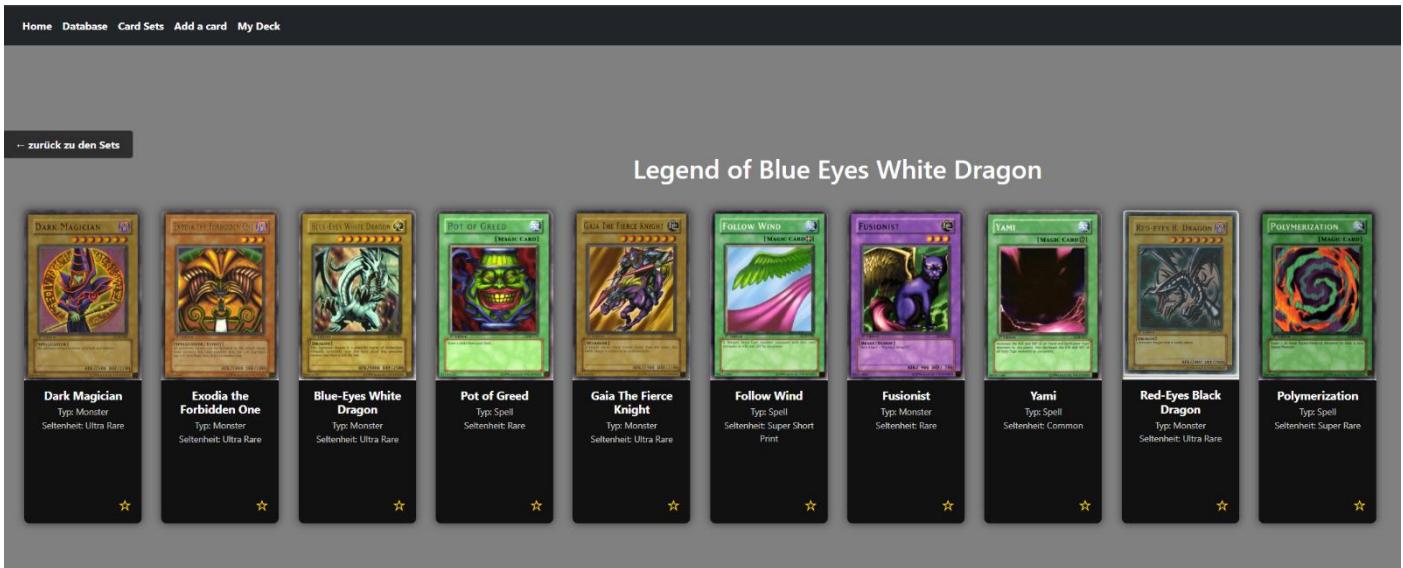
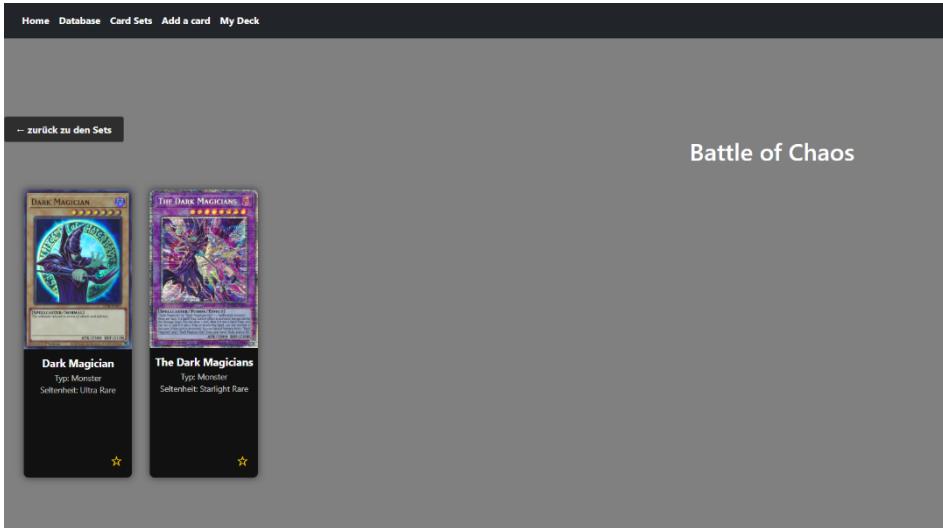
Auf dieser Seite werden verschiedene Yu-Gi-Oh!-Kartensets angezeigt. Die Sets stammen aus der MongoDB-Datenbank und werden in einer übersichtlichen Grid-Darstellung mit stilisierten Boxen präsentiert. Jeder Set-Name ist als Link gestaltet – ein Klick darauf führt zur Detailansicht mit allen Karten des jeweiligen Sets. Die Darstellung erfolgt über eine responsive Grid-Komponente cardset-grid. Beim Überfahren eines Sets mit der Maus wird ein **Hover-Effekt** ausgelöst, der die Box leicht vergrößert und abdunkelt, um visuelles Feedback zu geben.

Dateien:

- lib/components/CardSetList.svelte
- routes/cardsets/+page.svelte
- routes/cardsets/+page.server.js
- lib/db.js

3.5 Card Sets (detaillierte Seite)

Route: /cardsets/[id]



Diese Seite zeigt alle Karten an, die zu einem bestimmten Yu-Gi-Oh! Kartenset gehören. Der Name des Sets wird als Überschrift dargestellt. Die Karten erscheinen in einem responsiven Grid und nutzen die Komponente CardItem.svelte zur Darstellung. Die gesamte Liste wird von CardList.svelte organisiert. Falls ein Set keine Karten enthält, wird eine entsprechende Nachricht angezeigt: „Keine Karten in diesem Set“. Ein Zurück-Button oberhalb der Seite führt zurück zur Kartensets-Übersicht. Die Daten werden serverseitig geladen, basierend auf der set_id aus der URL.

Dateien:

- lib/components/CardItem.svelte
- lib/components/CardList.svelte
- routes/cardsets/[id]/+page.svelte
- routes/cardsets/[id]/+page.server.js
- lib/db.js

3.6 Add a Card

Route: /cards/add

The screenshot shows a web browser window with the URL 'localhost:5173/cards/add'. The page title is 'Add a Card'. The form contains four input fields: 'Card Name' (with placeholder 'z.B. Dark Magician Girl'), 'Card Type' (with placeholder 'z.B. Spell/Monster'), 'Card Number' (with placeholder 'z.B. UVM-1200'), and 'Level' (a dropdown menu). Below the form is a blue 'Submit' button.

Auf dieser Seite können Nutzerinnen und Nutzer eine neue Yu-Gi-Oh!-Karte manuell erfassen und zur Datenbank hinzufügen.

Das Eingabeformular besteht aus folgenden Feldern:

- **Card Name** (Textfeld)
- **Card Type** (Textfeld)
- **Card Number** (Textfeld)
- **Level** (Dropdown von 1 bis 9)

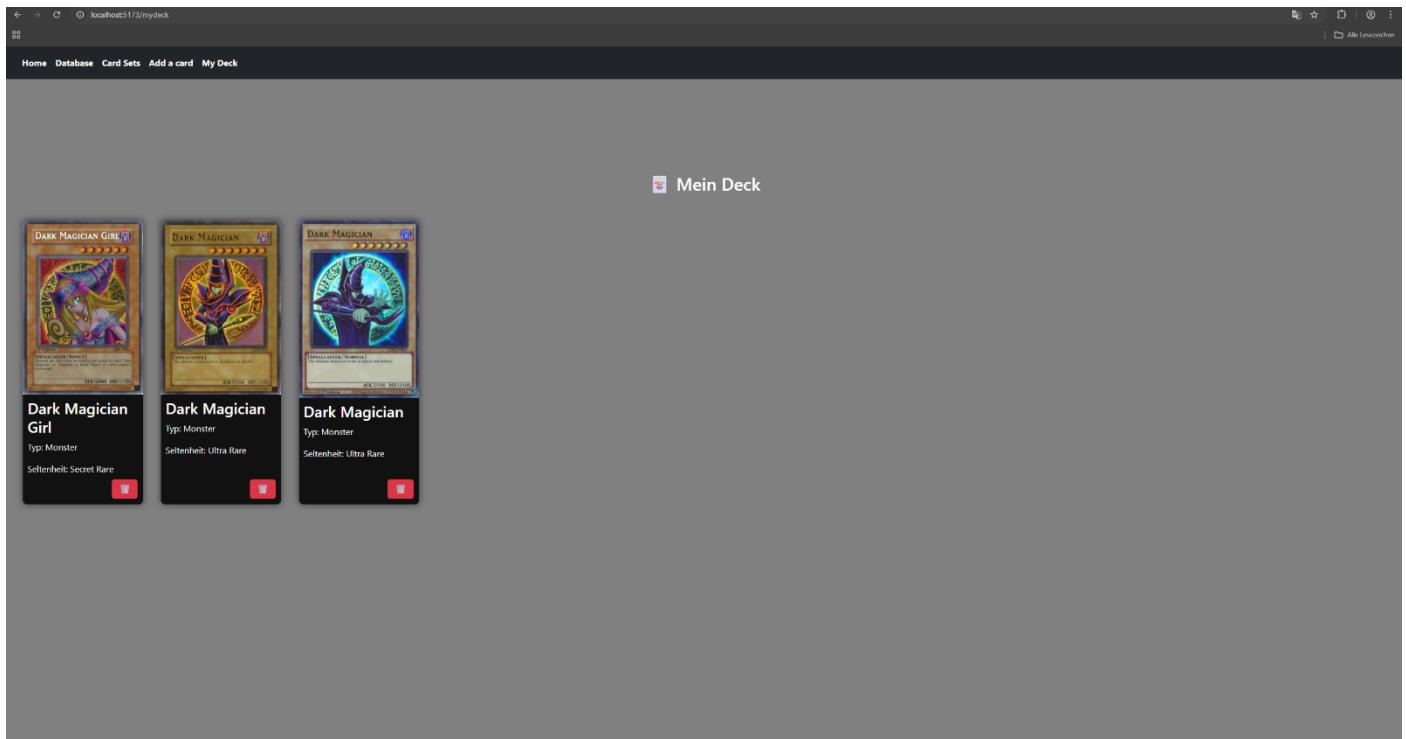
Nach dem Absenden wird die Karte mithilfe einer **Form Action** in der MongoDB gespeichert. Bei erfolgreicher Speicherung erscheint eine grüne Bestätigungsmeldung: „ **Karte erfolgreich gespeichert!**“. Falls ein Fehler auftritt, wird eine entsprechende Fehlermeldung angezeigt.

Dateien:

- lib/components/CardForm.svelte
- routes/add/+page.svelte
- routes/add/+page.server.js
- lib/db.js

3.7 My Deck

Route: /mydeck



Auf dieser Seite sehen Nutzerinnen und Nutzer alle Yu-Gi-Oh!-Karten, die sie ihrem persönlichen Deck hinzugefügt haben. Die Karten werden in einem Raster angezeigt und beinhalten **Bild**, **Name**, **Typ** und **Seltenheit**. Möchte man eine Karte aus dem Deck entfernen, genügt ein Klick auf das -Symbol. Falls sich noch keine Karten im Deck befinden, wird die Meldung angezeigt: „**Dein Deck ist leer**“.

Dateien:

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

4. Erweiterungen

4.1. Suchfunktion in der Kartenübersicht

In der Kartenübersicht auf /cards können Nutzerinnen und Nutzer gezielt nach bestimmten Karten suchen. Die Suchanfrage wird serverseitig verarbeitet und per Regex mit dem Kartennamen abgeglichen.

Dateien:

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

4.2. Favoritenfunktion

Karten können auf der Seite /cards als Favorit markiert werden (). Wird der Favoriten-Filter aktiviert, zeigt die Seite nur noch Karten, die entsprechend markiert wurden.

Dateien:

- routes/cards/+page.svelte
- lib/db.js

4.3. Responsives Design

Die gesamte Anwendung wurde responsiv gestaltet. Karten, Sets und Seitenlayouts passen sich automatisch an unterschiedliche Bildschirmgrößen an.

Dateien:

- lib/components/*.svelte
- routes/+layout.svelte
- static/app.css