

Prototyping Projektdokumentation

Name: Vince Käser

E-Mail: kaesevin@students.zhaw.ch

URL der deployten Anwendung: <https://gameworld1291.netlify.app/>

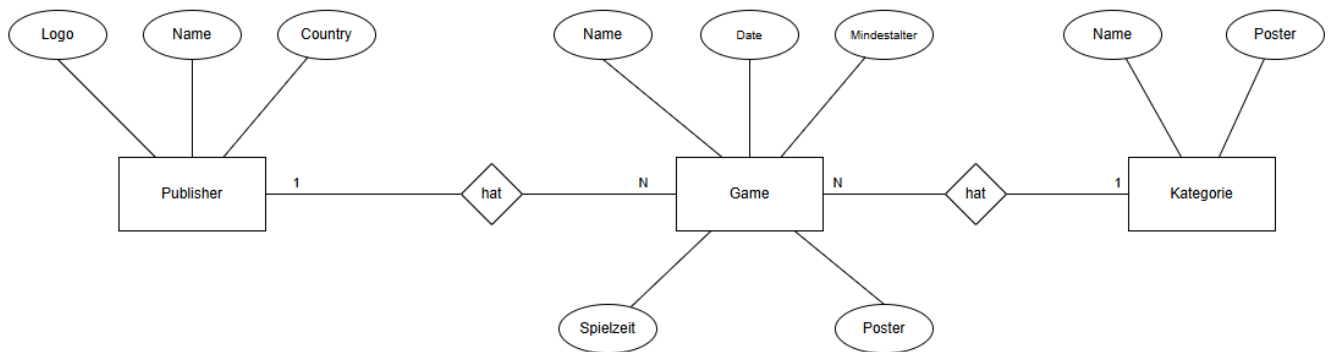
Einleitung

GameWorld wurde als zentrale Plattform entwickelt, die es Gamern ermöglicht, ihre Videospiele zu verwalten und zu organisieren. Der Fokus liegt dabei auf einer übersichtlichen Darstellung aller wichtigen Spielinformationen und einer intuitiven Verwaltung der persönlichen Spielebibliothek.

Die Kernfunktionalität der Webanwendung besteht in der Verwaltung von Spielen, die durch verschiedene Attribute wie Erscheinungsdatum, Spieldauer und Genre charakterisiert werden. Nutzer können diese Spiele ihrer persönlichen Bibliothek hinzufügen oder daraus entfernen. Eine besondere Stärke von GameWorld liegt in der Vernetzung der Daten: Spiele sind mit ihren jeweiligen Genres und Publishern verknüpft, wodurch Nutzer schnell verwandte Spiele finden oder das Portfolio bestimmter Publisher erkunden können.

Die Plattform strukturiert ihre Inhalte in drei Hauptbereiche: Die Spieleübersicht zeigt alle verfügbaren Spiele und ermöglicht das Hinzufügen neuer Titel. Der Genre-Bereich kategorisiert Spiele nach ihren Spielarten und erleichtert so die gezielte Suche nach bestimmten Spieltypen. Die Publisher-Sektion bietet detaillierte Informationen zu Spieleentwicklern und deren veröffentlichten Titeln. Diese durchdachte Strukturierung, kombiniert mit einem klaren Design im Dark Mode, macht GameWorld zu einer benutzerfreundlichen Plattform für alle Gaming-Enthusiasten.

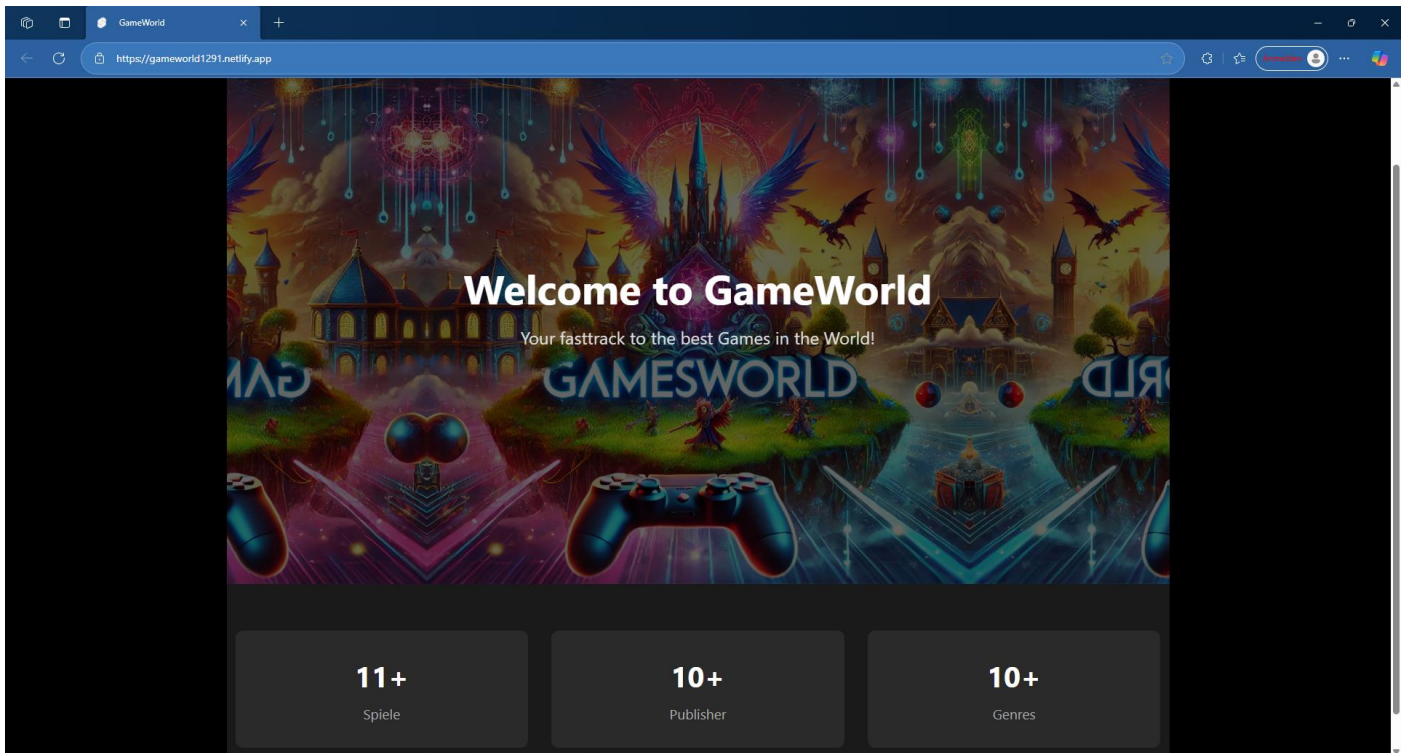
Datenmodell



Beschreibung der Anwendung

Hauptseite

Route: /



Ein animierter Willkommenstext "Welcome to GameWorld" erscheint durch einen Typewriter-Effekt, gefolgt von dem Untertitel "Your fasttrack to the best Games in the World". Unterhalb des Hero-Bereichs befindet sich eine Statistik-Section, die in drei Karten die aktuellen Zahlen der Plattform präsentiert. Diese Karten zeigen dynamisch die Gesamtanzahl der Spiele, Publisher und Genres aus der Datenbank an. Beim Überfahren mit der Maus werden die Karten durch einen Hover-Effekt leicht angehoben.

Dateien:

- /routes/+page.svelte

Erweiterungen

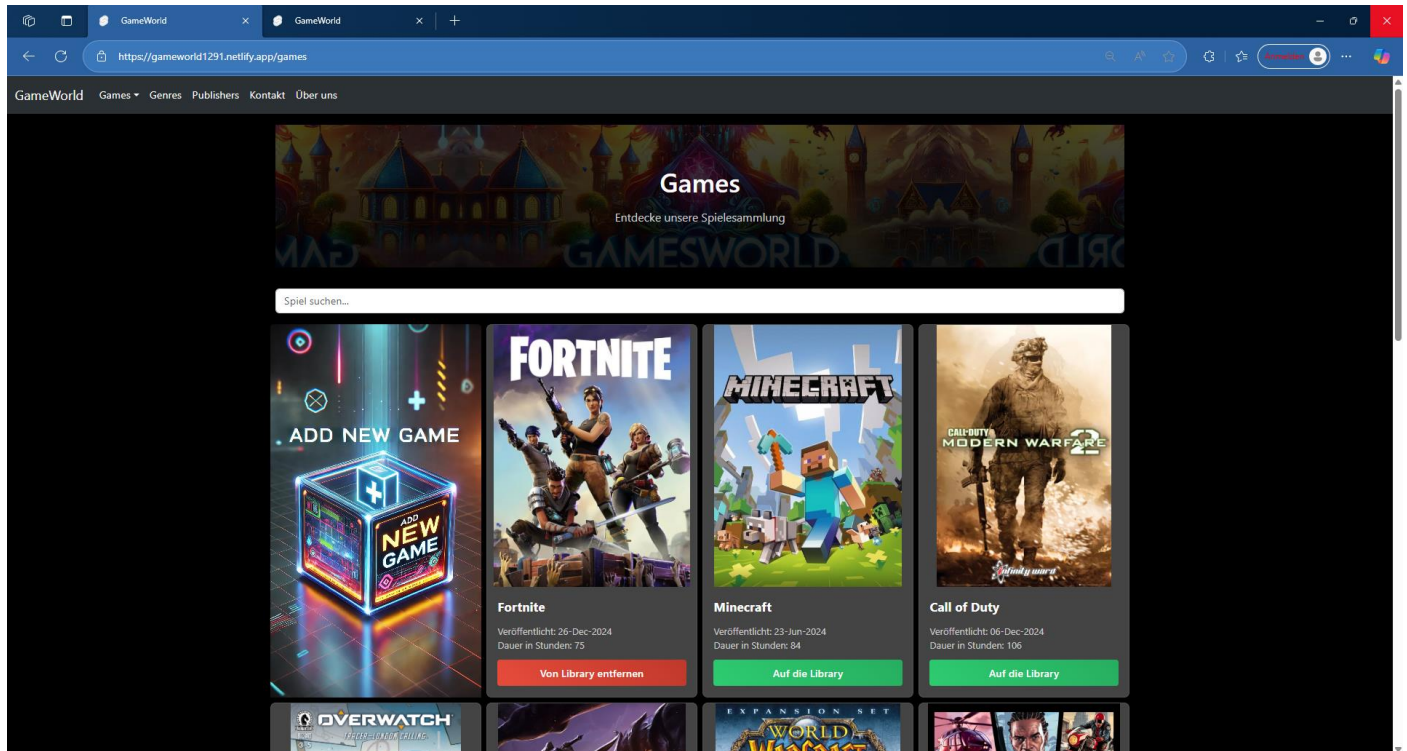
- Animierter Typewriter-Effekt für die Hauptüberschrift
- Fade-In Animation für den Untertitel
- Dynamische Statistik-Anzeige mit Echtzeit-Daten aus der Datenbank
- Interaktive Hover-Effekte bei den Statistik-Karten
- Integration eines grossflächigen Hero-Bereichs mit Hintergrundbild
- Responsives Design für optimale Darstellung auf allen Geräten

Dateien:

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

Game-Übersicht

Route: /games



Die Games-Übersicht präsentiert sich mit einem Hero-Banner, der den Titel "Games" und den Untertitel "Entdecke unsere Spielesammlung" enthält. Die Hauptfunktion dieser Seite ist die Darstellung aller verfügbaren Spiele in einem Raster-Layout. Jedes Spiel wird durch eine GameCard repräsentiert, die das Cover-Bild, den Titel, das Erscheinungsdatum und die Spieldauer anzeigt. Eine spezielle Card zum Hinzufügen neuer Spiele ist permanent am Anfang des Rasters platziert - diese zeigt beim Hover ein "+" Symbol. Bei jedem Spiel gibt es die Möglichkeit, dieses über einen Button zur persönlichen Library hinzuzufügen oder daraus zu entfernen. Mit einer Suchfunktion kann zudem nach jedem Game einzeln gefiltert werden

Dateien:

- /routes/games/+page.svelte
- /routes/games/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js

Erweiterungen

- Implementierung eines Grid-Layouts mit responsivem Design
- Dynamische "Add Game" Card mit Hover-Effekt
- Gestaltung der GameCards mit Hover-Animationen
- Integration eines Hero-Banners für bessere visuelle Strukturierung
- Optimierte Darstellung der Spiel-Informationen in den Cards
- Suchleiste zum Suchen nach einzelnen Games

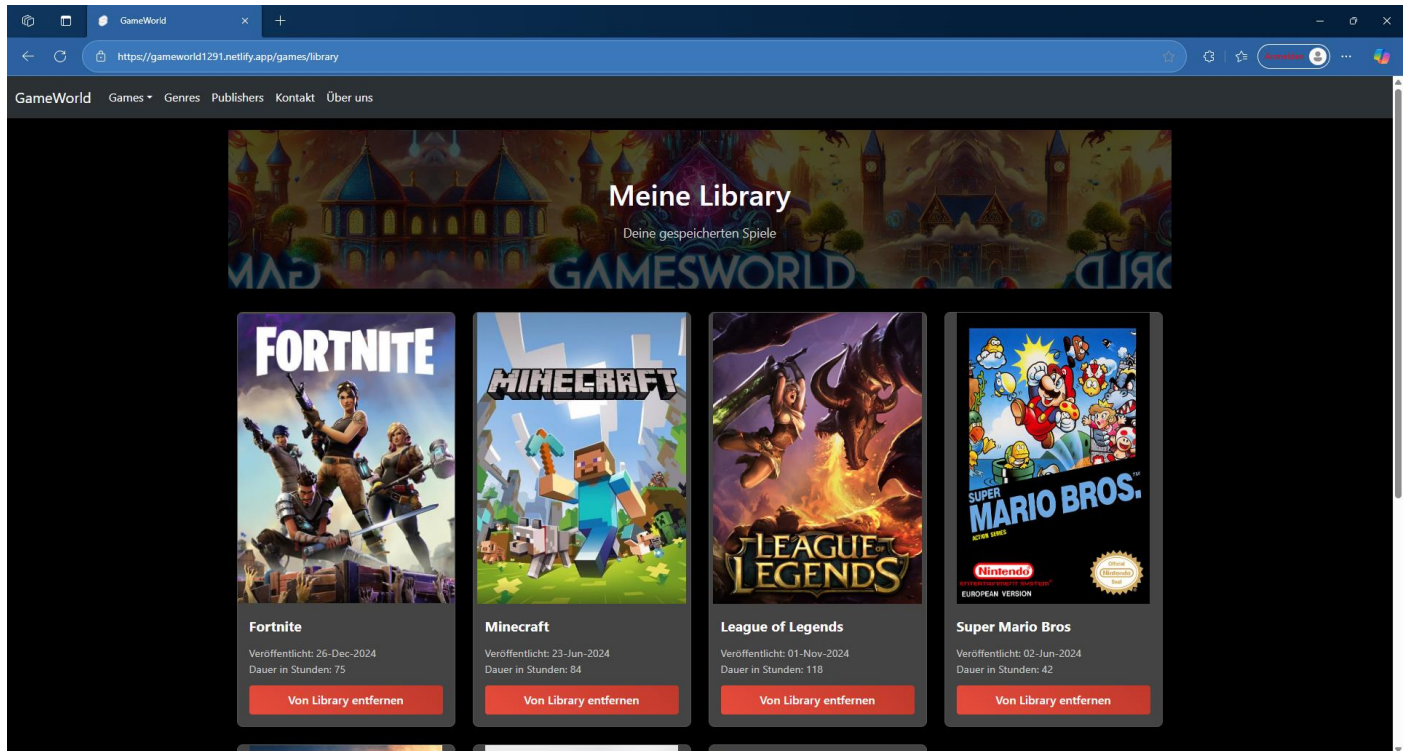
Dateien:

- /routes/games/+page.svelte

- /routes/games/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js

Libraryseite

Route: /games/library



Die Library-Seite folgt dem gleichen Design-Prinzip wie die Games-Übersicht, beginnend mit einem Hero-Banner mit dem Titel "Library". Sie zeigt ausschliesslich die Spiele an, die zuvor vom Benutzer zur Library hinzugefügt wurden. Die Spiele werden ebenfalls im Raster-Layout mit GameCards dargestellt, jedoch erscheint hier keine "Add Game" Card. Jede GameCard zeigt das Cover-Bild, den Titel, das Erscheinungsdatum und die Spieldauer des Spiels an. Über den "Von Library entfernen" Button können Spiele direkt aus der persönlichen Sammlung entfernt werden.

Dateien:

- /routes/games/library/+page.svelte
- /routes/games/library/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js

Erweiterungen

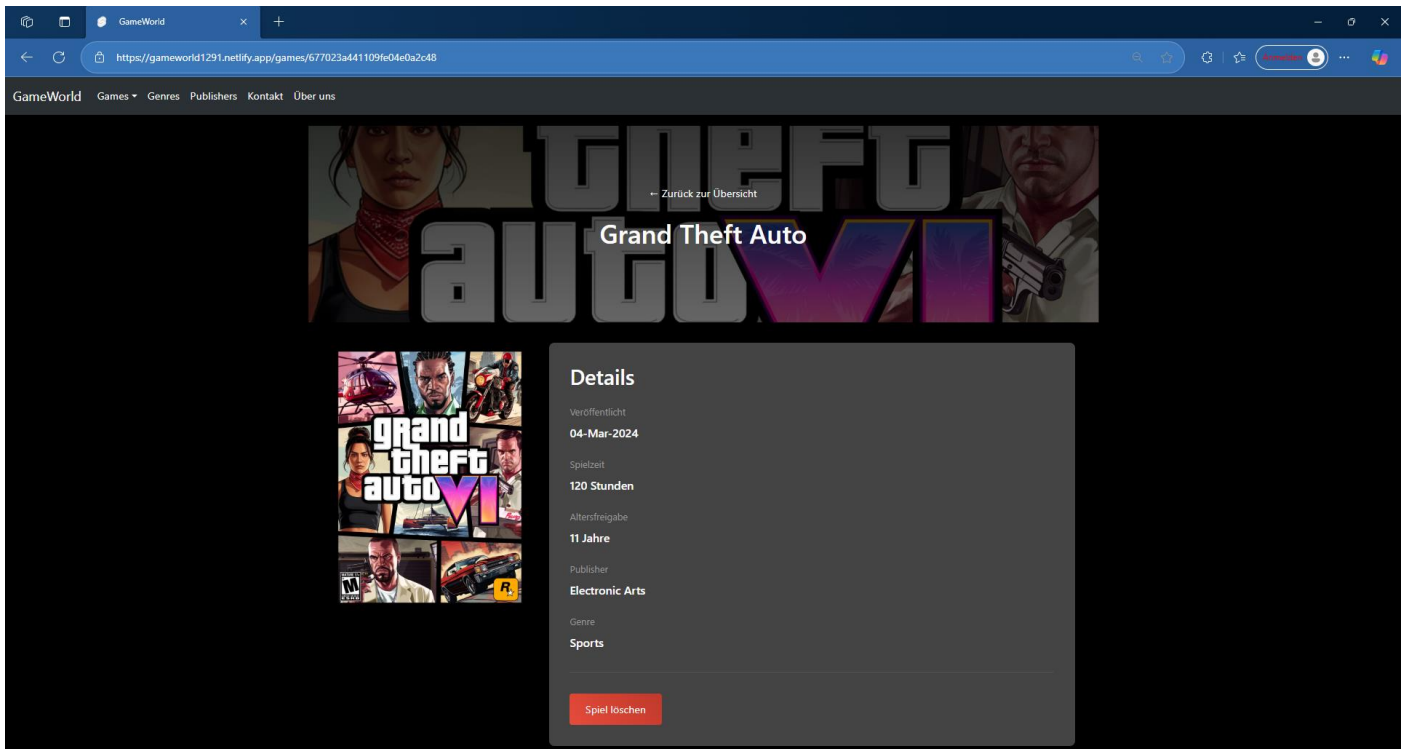
- Filterlogik zur Anzeige der Library-Spiele
- Wiederverwendung der GameCard-Komponente mit angepasster Funktionalität
- Integration eines spezifischen Hero-Banners für die Library
- Responsives Grid-Layout für optimale Darstellung auf allen Geräten
- Dynamische Aktualisierung der Ansicht: Spiele verschwinden sofort nach dem Entfernen aus der Library, ohne dass die Seite neu geladen werden muss
- Extraseite für die Library

Dateien:

- /routes/games/library/+page.svelte
- /routes/games/library/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js

Game-Detailseite

Route: /games/[game_id]



Die Detailseite eines einzelnen Spiels beginnt mit einem grossen Hero-Banner, der das Cover-Bild des jeweiligen Spiels als Hintergrundbild verwendet. Darüber liegt ein dezenter Verdunklungseffekt, auf dem der "Zurück" Button und der Spieltitel angezeigt werden. Die verknüpften Daten werden durch die Funktionen `getGenreByGame()` und `getPublisherByGame()` bereitgestellt. Unterhalb des Hero-Bereichs ist die Seite in zwei Spalten aufgeteilt: Links wird das Cover-Bild des Spiels nochmals in normaler Grösse angezeigt, rechts befinden sich die detaillierten Informationen. Diese umfassen das Veröffentlichungsdatum, die Spieldauer, sowie Verknüpfungen zum zugehörigen Genre und Publisher. Die Genre- und Publisher-Namen sind als Links gestaltet und führen zu den jeweiligen Übersichtsseiten. Am Ende der Informationssektion befindet sich ein "Delete" Button zum Löschen des Spiels.

Dateien:

- /routes/games/[game_id]/+page.svelte
- /routes/games/[game_id]/+page.server.js
- /lib/db.js

Erweiterungen

- Dynamischer Hero-Banner mit dem jeweiligen Spiel-Cover als Hintergrundbild
- Zweispaltiges Layout für optimale Informationsdarstellung
- Verknüpfung der Genre- und Publisher-Informationen mit den entsprechenden Detailseiten
- Stilvoller "Zurück" Button im einheitlichen Design

- Responsives Design: Anpassung des Layouts für mobile Geräte
- Implementierung einer Löschfunktion mit direkter Weiterleitung zur Spieleübersicht
- Dynamische Datenladelogik für verknüpfte Genre- und Publisher-Informationen
- `getGenreByGame()` und `getPublisherByGame()` Funktionen für MongoDB Abfragen

Dateien:

- `/routes/games/[game_id]/+page.svelte`
- `/routes/games/[game_id]/+page.server.js`
- `/lib/db.js`

Spiel hinzufügen

Route: `/games/create`

The screenshot shows a web browser window with the URL `https://gameworld1291.netlify.app/games/create`. The page has a navigation bar with links: GameWorld, Games, Genres, Publishers, Kontakt, Über uns. The main content area is titled 'Add a Game' with a 'Zurück' button. The form contains the following fields:

- Name:
- Veröffentlicht: (with a validation message: 'Füllen Sie dieses Feld aus.')
- Spielzeit in Stunden:
- Mindestalter:
- Publisher: (with 'Electronic Arts' selected)
- Kategorie: (with 'Action' selected)
- Spiel-Cover: (with 'Datei auswählen' and 'Keine Datei ausgewählt' options)

An 'Add Game' button is located at the bottom of the form.

Die "Create Game" Seite ermöglicht das Hinzufügen neuer Spiele zur Datenbank. Die Seite beginnt mit einem "Zurück" Button und der Überschrift "Add a Game". Das Hauptelement ist ein Formular mit folgenden Pflichtfeldern: Name des Spiels, Veröffentlichungsdatum, Spielzeit in Stunden und Mindestalter. Zusätzlich enthält das Formular zwei Dropdown-Menüs für die Auswahl des Publishers und des Genres, deren Optionen durch die `getPublishers()` und `getGenres()` Funktionen geladen werden. Darüber hinaus besteht die Möglichkeit ein Bild des Spiel Covers einzufügen. Nach erfolgreichem Erstellen eines Spiels wird ein Popup-Fenster angezeigt, das den Benutzer über die erfolgreiche Erstellung informiert.

Dateien:

- `/routes/games/create/+page.svelte`
- `/routes/games/create/+page.server.js`
- `/lib/db.js`
- `/routes/styles.css`

Erweiterungen:

- Entwicklung eines übersichtlichen Formularlayouts mit Pflichtfeld-Validierung

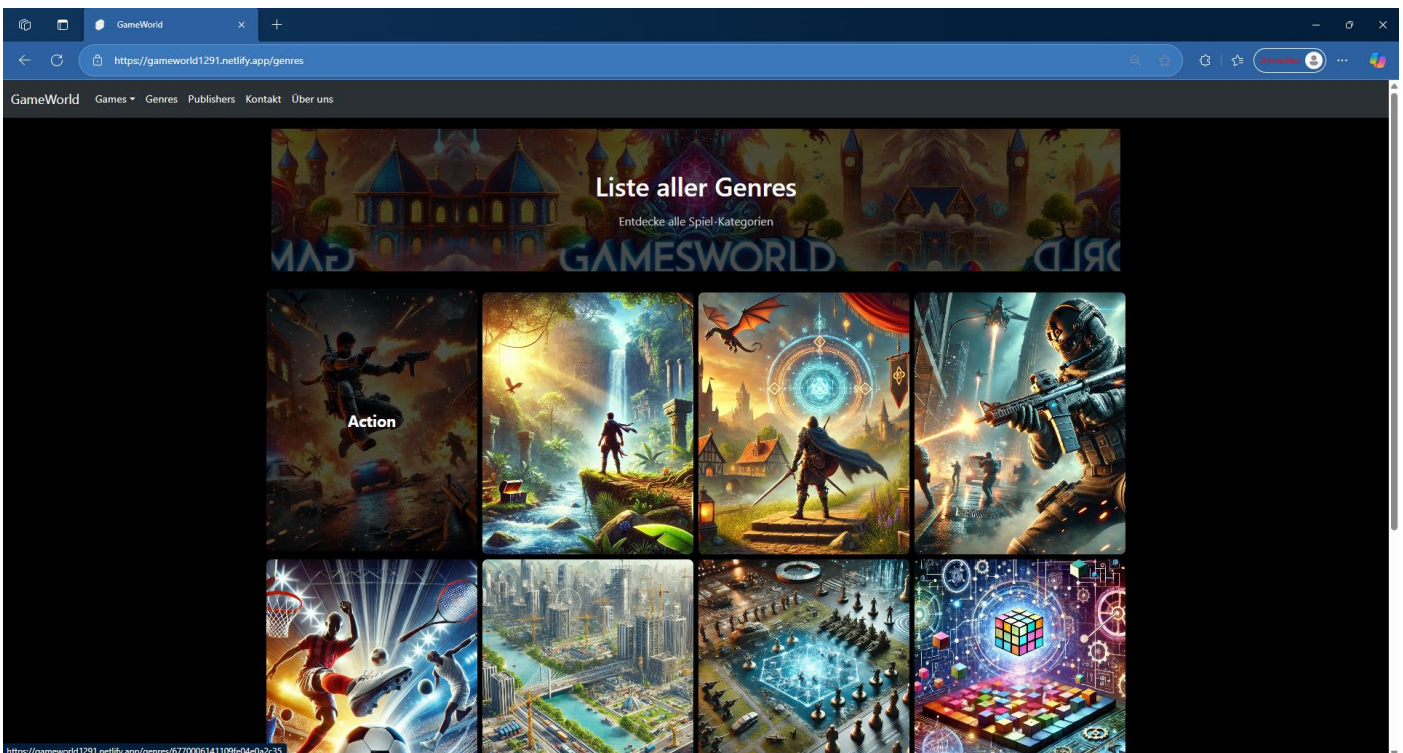
- Integration von dynamischen Dropdown-Menüs für Publisher und Genre
- Popup-Fenster bei erfolgreichem Erstellen eines Games
- Möglichkeit ein Bild als Cover einzufügen

Dateien:

- /routes/games/create/+page.svelte
- /routes/games/create/+page.server.js
- /lib/db.js
- /routes/styles.css

Genre-Übersicht

Route: /genres



Die Genre-Übersichtsseite startet mit einem Hero-Banner, der den Titel "Liste aller Genres" und den Untertitel "Entdecke alle Spiel-Kategorien" präsentiert. Die Genres werden in einem responsiven Grid-Layout dargestellt. Jedes Genre wird durch eine GenreCard repräsentiert, die das vollflächige Genre-Bild zeigt. Beim Überfahren einer Genre-Karte mit der Maus erscheint ein eleganter Overlay-Effekt mit einem halbtransparenten dunklen Hintergrund, der den Genre-Namen einblendet. Die gesamte Karte ist als Link zur Genre-Detailseite gestaltet und hebt sich beim Hover leicht an. Jede GenreCard fungiert als interaktives Element, das den Nutzer zur spezifischen Genre-Seite mit allen zugehörigen Spielen führt.

Dateien:

- /routes/genres/+page.svelte
- /routes/genres/+page.server.js
- /lib/components/GenreCard.svelte
- /lib/db.js

- /routes/styles.css

Erweiterungen

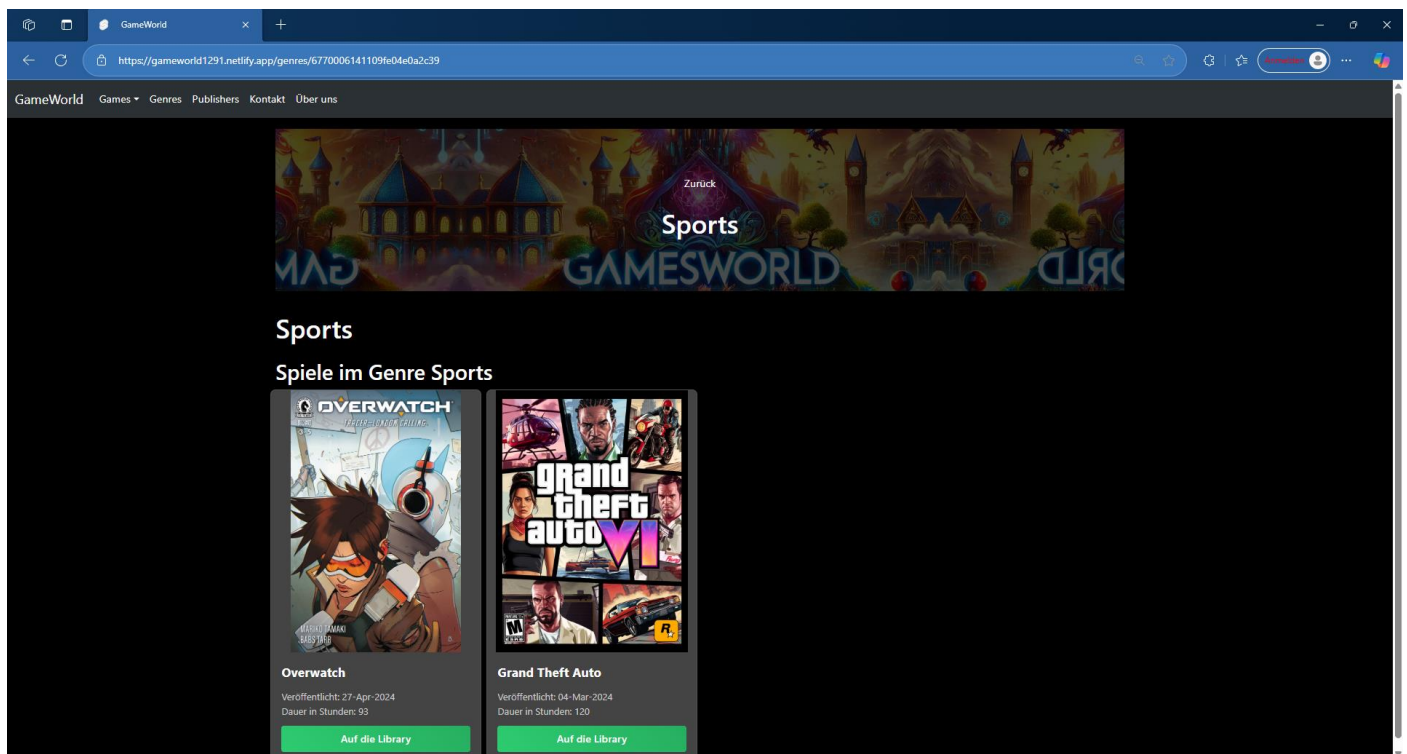
- Implementierung eines modernen Grid-Layouts für die Genre-Darstellung
- Entwicklung einer spezialisierten GenreCard-Komponente mit Hover-Effekten
- Bildbasierte Darstellung der Genres mit eleganten Overlay-Effekten
- Animation der Karten beim Hover (Anheben)
- Optimierte Bilddarstellung mit einheitlicher Grösse und Ausrichtung

Dateien:

- /routes/genres/+page.svelte
- /routes/genres/+page.server.js
- /lib/components/GenreCard.svelte
- /lib/db.js
- /routes/styles.css

Genre-Detailseite

Route: /genres/[genre_id]



Die Genre-Detailseite zeigt alle Spiele einer spezifischen Genre-Kategorie. Ein Hero-Banner am Anfang der Seite enthält einen "Zurück" Button zur Genre-Übersicht und präsentiert den Namen des ausgewählten Genres. Unterhalb des Banners werden alle zum Genre gehörenden Spiele in einem Grid-Layout angezeigt, die durch die Funktion `getGamesByGenre()` abgerufen werden. Die Spiele werden dabei mit der gleichen GameCard-Komponente wie in der allgemeinen Spieleübersicht dargestellt, inklusive Cover-Bild, Titel, Veröffentlichungsdatum, Spieldauer und der Möglichkeit, das Spiel zur Library hinzuzufügen oder daraus zu entfernen.

Dateien:

- /routes/genres/[genre_id]/+page.svelte
- /routes/genres/[genre_id]/+page.server.js

- /lib/components/GameCard.svelte
- /lib/db.js
- /routes/styles.css

Erweiterungen

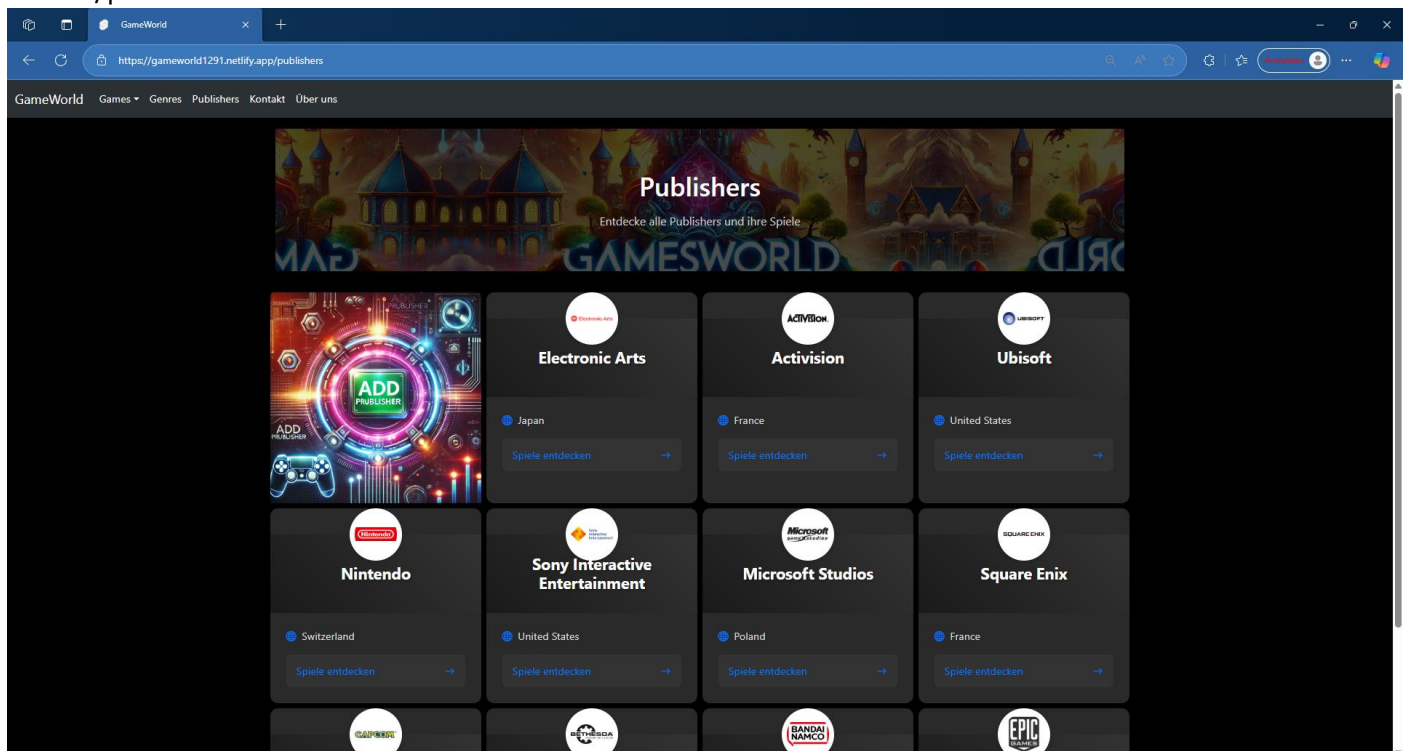
- Komplette Seite ist eine Erweiterung
- Filterung und Anzeige von Spielen nach spezifischem Genre
- Wiederverwendung der GameCard-Komponente für konsistentes Design
- Integration des Hero-Banners mit Genre-spezifischen Informationen
- Implementierung der "Zurück" Navigation
- Responsives Grid-Layout für die Spieledarstellung
- Dynamisches Laden der genrespezifischen Spiele-Daten
- `getGamesByGenre()` Funktion in `db.js` für MongoDB Abfrage

Dateien:

- /routes/genres/[genre_id]/+page.svelte
- /routes/genres/[genre_id]/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js
- /routes/styles.css

Publisher-Übersicht

Route: /publishers



Die Publisher-Übersichtsseite beginnt mit einem Hero-Banner, der den Titel "Publishers" und den Untertitel "Entdecke alle Publishers und ihre Spiele" anzeigt. Die Publisher werden in einem vierpaltigen Grid-Layout präsentiert. Jeder Publisher wird durch eine PublisherCard dargestellt, die ein charakteristisches Design aufweist: Ein

runder, weisser Container präsentiert zentral das offizielle Logo des jeweiligen Publishers. Dieser Logo-Container schwebt über einem dunkel gradierten Banner, in dem der Publisher-Name zentriert dargestellt wird. Unterhalb des Banners werden das Herkunftsland des Publishers und ein "Spiele entdecken" Button angezeigt. Beim Überfahren mit der Maus hebt sich die Karte leicht an und der "Spiele entdecken" Button zeigt eine dezente Animation des Pfeil-Icons. Eine spezielle Card zum Hinzufügen neuer Publisher ist permanent am Anfang des Rasters platziert - diese zeigt beim Hover ein "+" Symbol.

Dateien:

- /routes/publishers/+page.svelte
- /routes/publishers/+page.server.js
- /lib/components/PublisherCard.svelte
- /lib/db.js
- /routes/styles.css

Erweiterungen

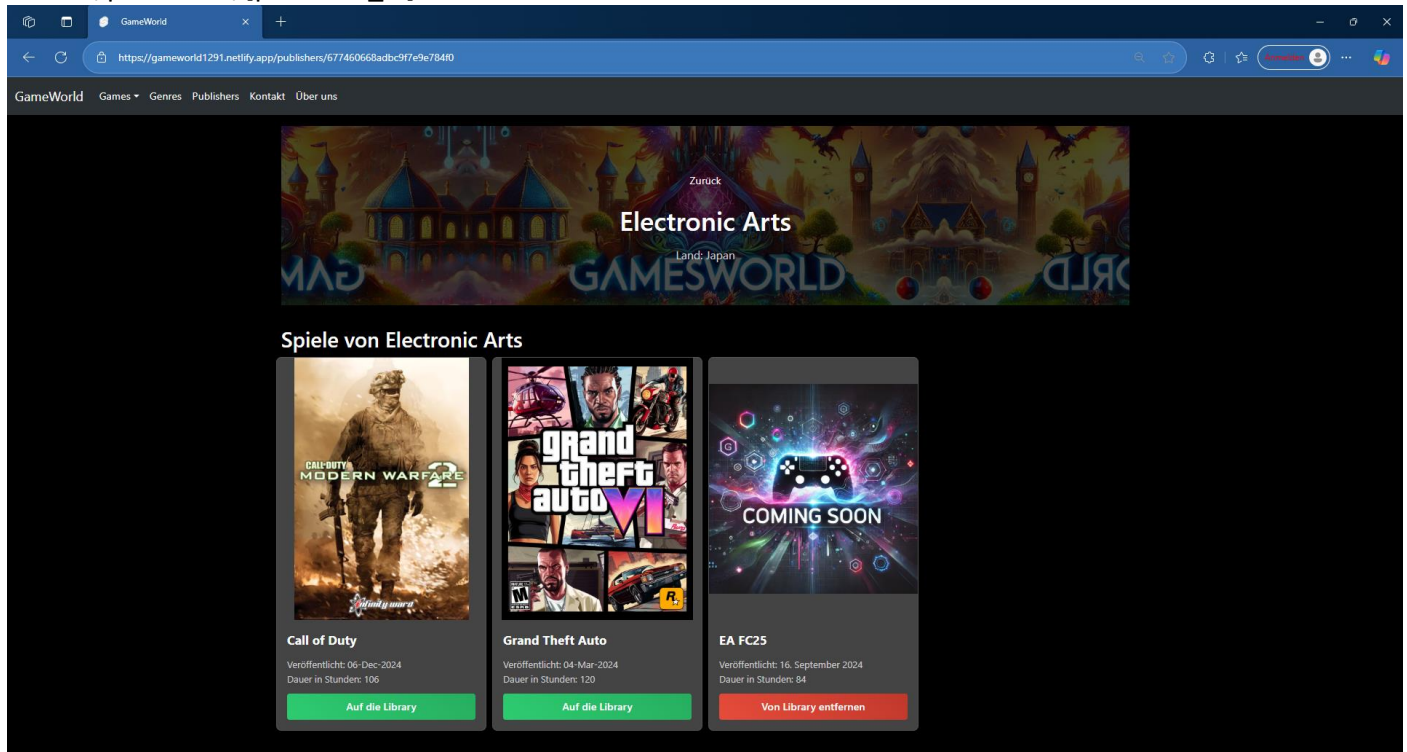
- Entwicklung einer spezialisierten PublisherCard-Komponente mit elegantem Design
- Integration eines schwebenden Logo-Containers für die Publisher-Logos mit weissem Hintergrund
- Animierte Hover-Effekte für die Publisher-Karten
- Interaktive Button-Animation mit Pfeil-Icon
- Einheitliches Design-Schema für Publisher-Informationen
- Dynamische "Add Publisher" Card mit Hover-Effekt

Dateien:

- /routes/publishers/+page.svelte
- /routes/publishers/+page.server.js
- /lib/components/PublisherCard.svelte
- /lib/db.js
- /routes/styles.css

Publisher-Detailseite

Route: /publishers/[publisher_id]



Die Publisher-Detailseite präsentiert im Hero-Banner den Namen des Publishers, das zugehörige Land und einen "Zurück" Button zur Publisher-Übersicht. Das Hintergrundbild des Banners ist das GameWorld-Banner, welches bei allen Hero-Bereichen verwendet wird. Unterhalb des Banners erscheint der Titel "Spiele von [Publisher Name]", gefolgt von einer Grid-Ansicht aller Spiele des ausgewählten Publishers, die mittels `getGamesByPublisher()` geladen werden. Die Spiele werden mit der GameCard-Komponente dargestellt, die das Cover-Bild, den Titel, das Veröffentlichungsdatum und die Spieldauer anzeigt. Jede GameCard bietet auch die Möglichkeit, das Spiel zur Library hinzuzufügen oder daraus zu entfernen.

Dateien:

- /routes/publishers/[publisher_id]/+page.svelte
- /routes/publishers/[publisher_id]/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js
- /routes/styles.css

Erweiterungen:

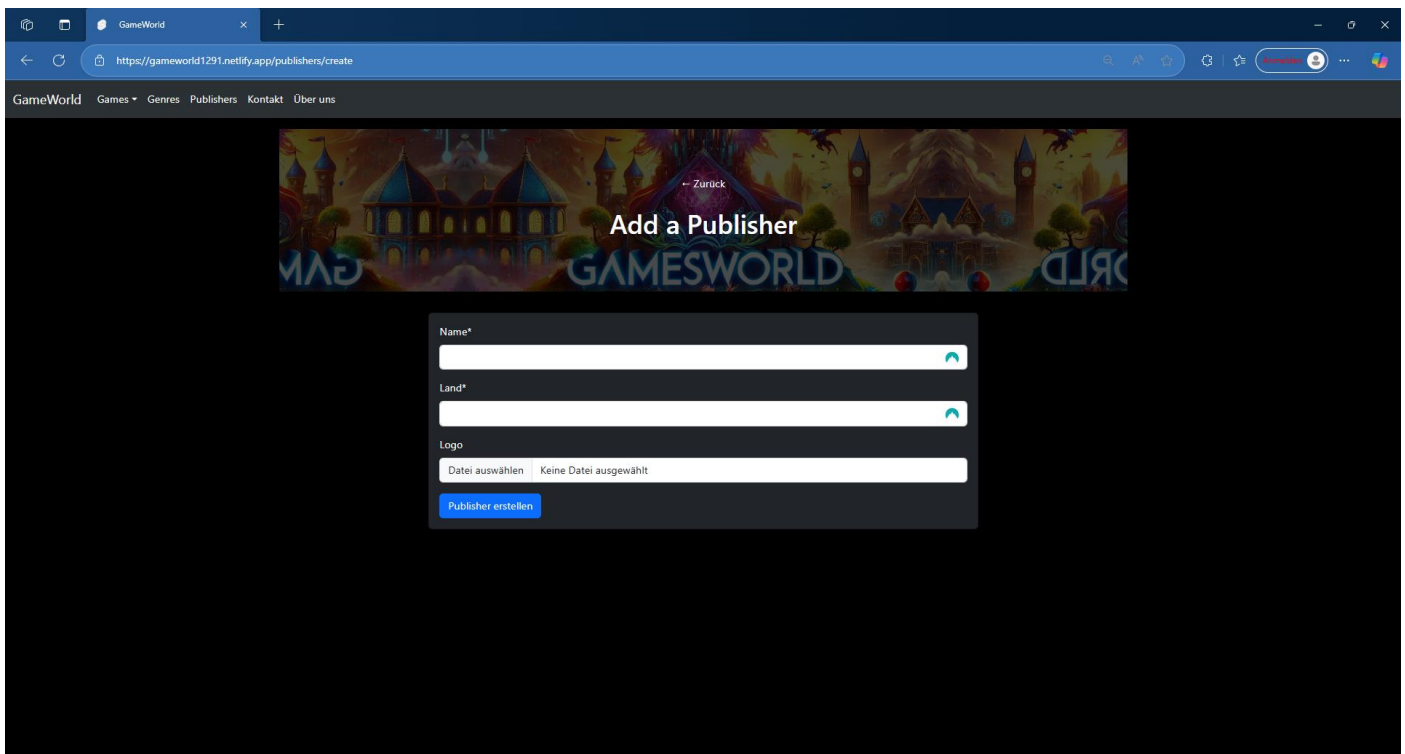
- Filterung und Anzeige von Spielen nach spezifischem Publisher
- Wiederverwendung der GameCard-Komponente für konsistentes Design
- Integration des Hero-Banners mit Publisher-spezifischen Informationen
- Implementierung der "Zurück" Navigation
- Responsives Grid-Layout für die Spieldarstellung
- Dynamisches Laden der publisherspezifischen Spiele-Daten
- `getGamesByPublisher()` Funktion in `db.js` für MongoDB Abfrage

Dateien:

- /routes/publishers/[publisher_id]/+page.svelte
- /routes/publishers/[publisher_id]/+page.server.js
- /lib/components/GameCard.svelte
- /lib/db.js
- /routes/styles.css

Create Publisher

Route: /publishers/create

The screenshot shows a web browser window with the URL 'https://gameworld1291.netlify.app/publishers/create'. The page has a dark theme and a header with a navigation menu: 'GameWorld', 'Games', 'Genres', 'Publishers', 'Kontakt', and 'Über uns'. The main content area features a large, colorful illustration of a fantasy city with the text 'Add a Publisher' and a 'Zurück' button. Below the illustration is a form with three fields: 'Name*' (required), 'Land*' (required), and 'Logo'. The 'Logo' field has a file selection button and a text input. At the bottom of the form is a blue button labeled 'Publisher erstellen'.

Die "Create Publisher" Seite ermöglicht das Hinzufügen neuer Publisher zur Datenbank. Die Seite beginnt mit einem "Zurück" Button und der Überschrift "Add a Publisher". Das Hauptelement ist ein Formular mit folgenden Pflichtfeldern: Name des Publishers und dem Herkunftsland. Darüber hinaus besteht die Möglichkeit ein Bild des Publishers einzufügen. Nach erfolgreichem Erstellen eines Publishers wird ein Popup-Fenster angezeigt, das den Benutzer über die erfolgreiche Erstellung informiert und zurück zur Publisher Hauptseite führt.

Dateien:

- /routes/publishers/create/+page.svelte
- /routes/publishers/create/+page.server.js
- /lib/db.js
- /routes/styles.css

Erweiterungen:

- Die komplette Seite ist eine Erweiterung
- Entwicklung eines übersichtlichen Formularlayouts mit Pflichtfeld-Validierung
- Popup-Fenster bei erfolgreichem Erstellen eines Games
- Möglichkeit ein Bild als Logo des Publishers einzufügen

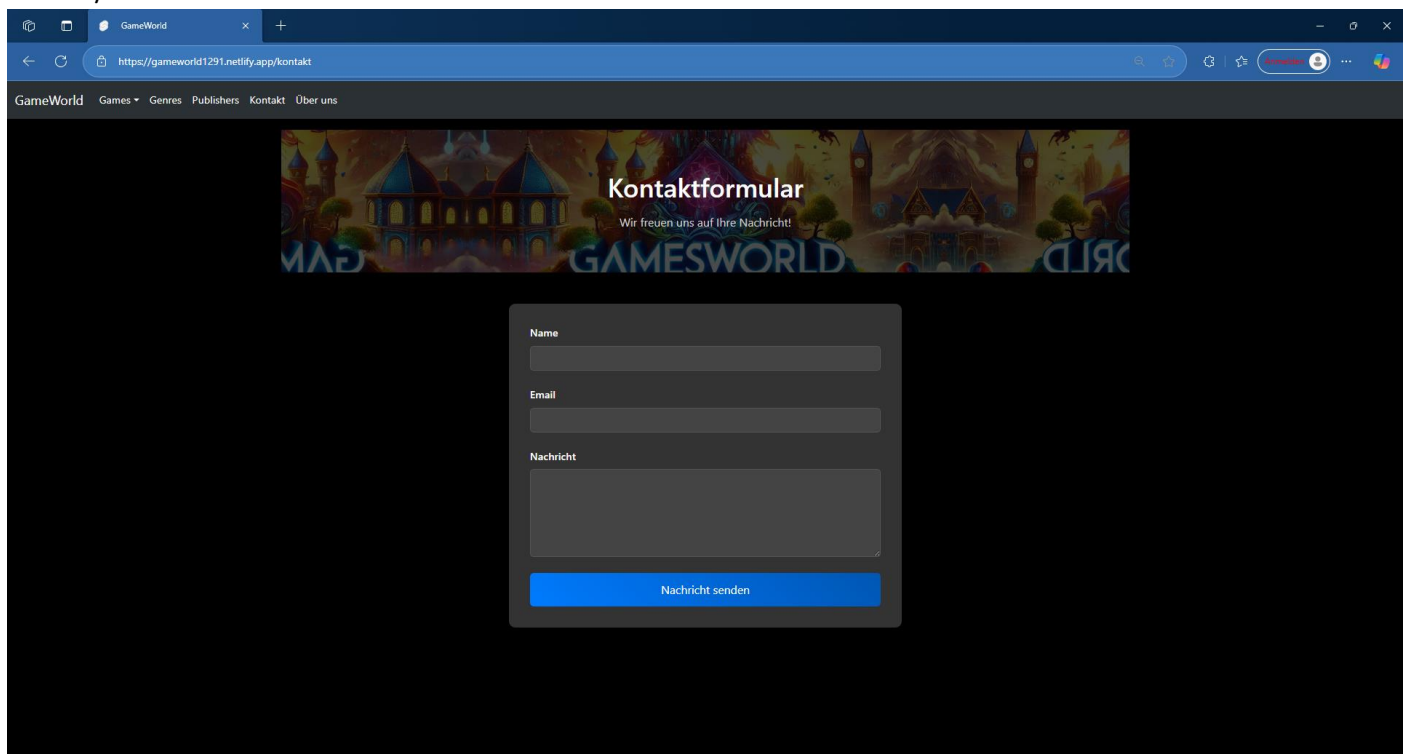
- `getPublisher()` Funktion in `db.js`
- Einheitlicher Hero-Banner mit Titel

Dateien:

- `/routes/publishers/create/+page.svelte`
- `/routes/publishers/create/+page.server.js`
- `/lib/db.js`
- `/routes/styles.css`

Kontaktseite

Route: `/kontakt`



Die Kontaktseite präsentiert sich mit einem Hero-Banner, der den Titel "Kontaktformular" enthält. Darunter befindet sich ein zentriertes, dunkles Formular-Panel mit eingegrenzter Breite. Das Formular besteht aus drei Hauptfeldern: Name, E-Mail und Nachricht. Alle Eingabefelder sind im dunklen Design gehalten mit helleren Hover- und Fokus-Effekten. Unter dem Nachrichtenfeld befindet sich ein grosser "Absenden" Button. Das Formular bietet Validierung der Eingaben und zeigt entsprechendes Feedback an. Nach erfolgreichem Absenden erscheint eine Erfolgsmeldung oberhalb des Formulars.

Dateien:

- `/routes/kontakt/+page.svelte`
- `/routes/styles.css`

Erweiterungen

- Die komplette Seite ist eine Erweiterung
- Entwicklung eines ansprechenden, dunklen Formular-Designs
- Implementierung von Formularvalidierung (Name min. 2 Zeichen, gültige Email, Nachricht min. 10 Zeichen)
- Animierter Feedback-Effekte bei Hover und Fokus auf den Eingabefeldern

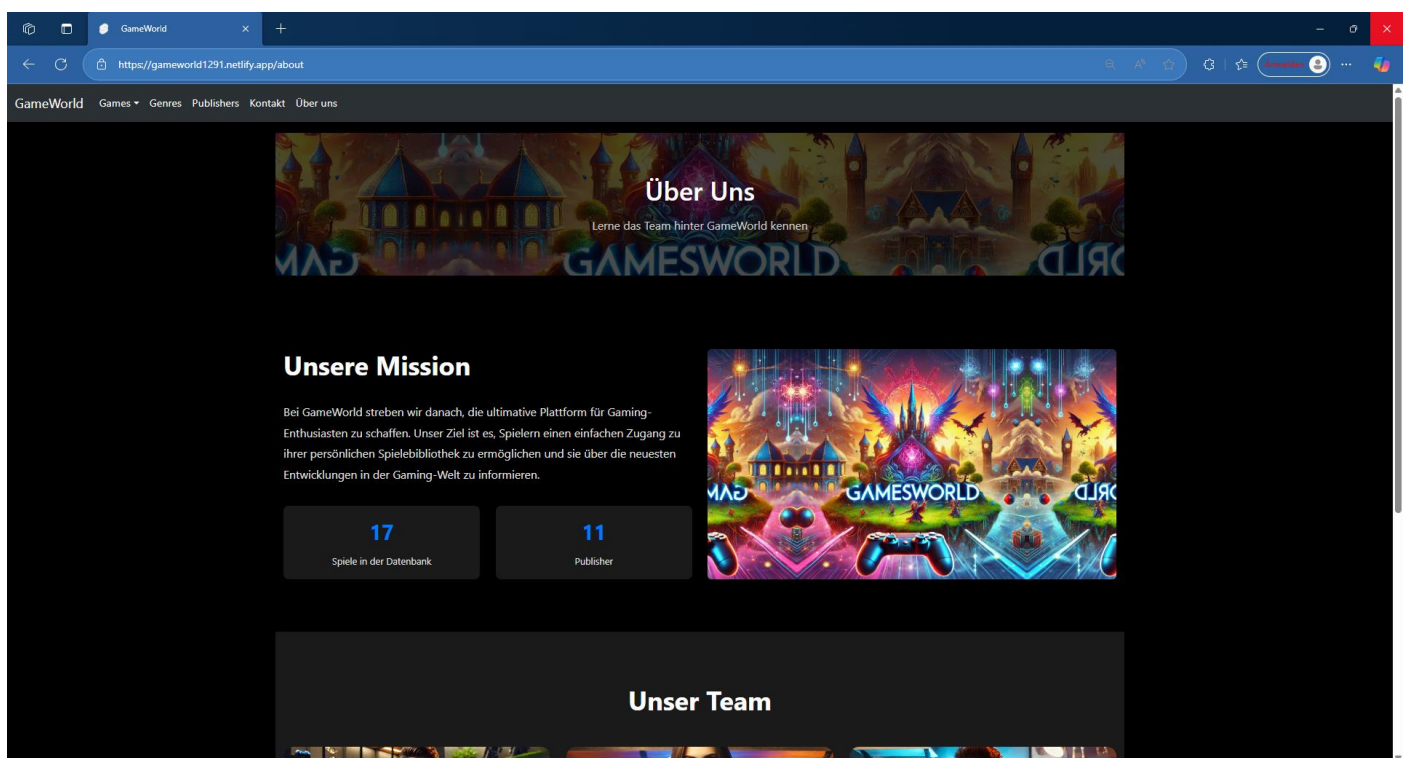
- Loading-Status während des Absendens mit Spinner-Animation
- Erfolgs- und Fehlermeldungen mit Animation
- Deaktivierung der Eingabefelder während des Sendevorgangs
- Einheitlicher Hero-Banner mit Titel und Untertitel

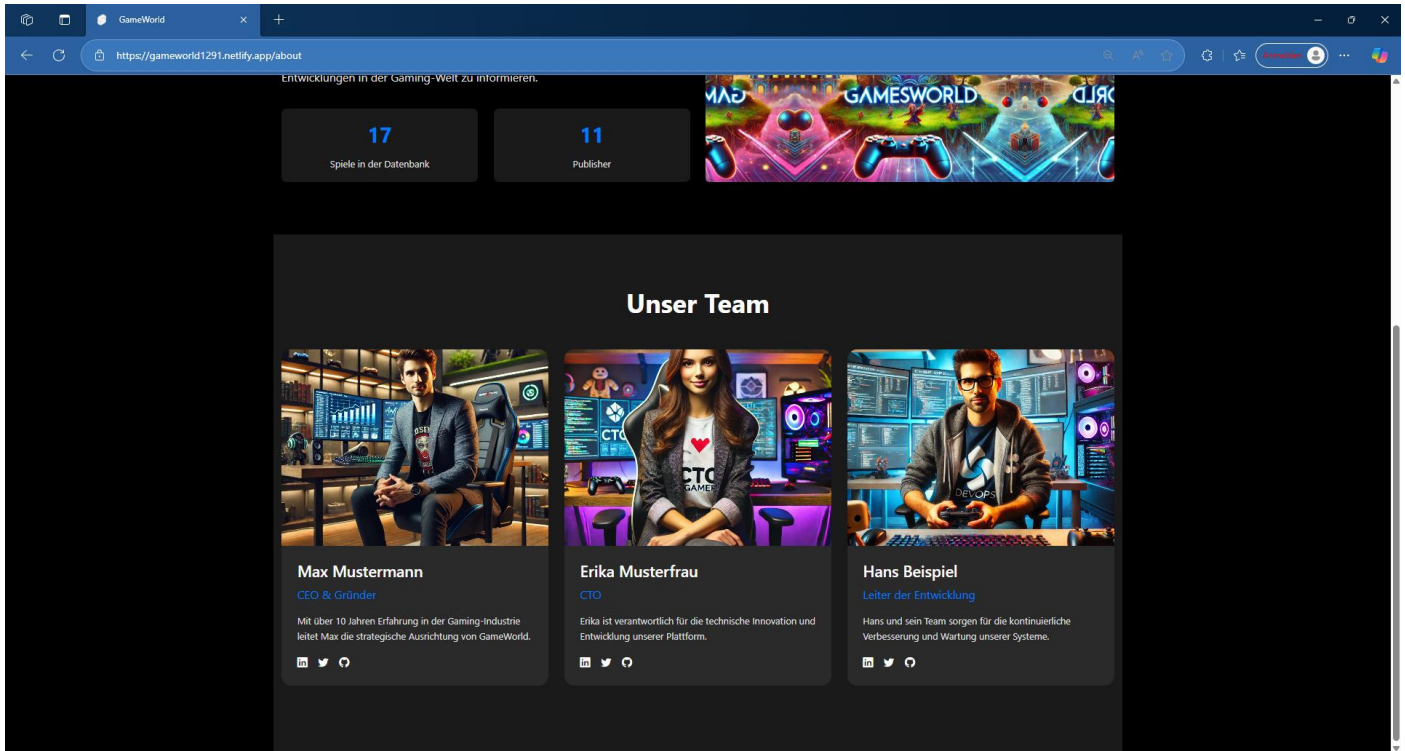
Dateien:

- /routes/kontakt/+page.svelte
- /routes/styles.css

Über uns

Route: /about





Die "Über Uns" Seite startet mit einem Hero-Banner, der den Titel "Über Uns" und den Untertitel "Erfahre mehr über unser Team" präsentiert. Darunter wird eine Statistik-Section mit dynamischen Zahlen aus der Datenbank angezeigt, die die aktuellen Plattform-Statistiken (Anzahl Spiele und Publisher) in animierten Karten darstellt. Der letzte Abschnitt zeigt eine Team-Übersicht, in der jedes Teammitglied mit Bild, Namen, Position und einer kurzen Beschreibung vorgestellt wird. Die Teammitglied-Karten enthalten zudem Platzhalter für Social Media Icons.

Dateien:

- `/routes/about/+page.svelte`
- `/routes/about/+page.server.js`
- `/lib/db.js`
- `/routes/styles.css`

Erweiterungen

- Die komplette Seite ist eine Erweiterung
- Integration der dynamischen Statistik-Anzeige
- Gestaltung der Team-Karten mit Hover-Effekten
- Einbindung von Bootstrap Icons für Social Media
- Dynamisches Laden der Plattform-Statistiken
- Einheitlicher Hero-Banner

Dateien:

- `/routes/about/+page.svelte`
- `/routes/about/+page.server.js`
- `/lib/db.js`
- `/routes/styles.css`