

Prototyping Projektdokumentation

Name: Miriam Holfelder

E-Mail: holfemir@students.zhaw.ch

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

1. Einleitung

MyArmory ist eine Webanwendung zur Verwaltung von World of Warcraft Charakteren und deren Gilden.

Nutzer:innen können ihre eigenen Charaktere importieren, Details einsehen, Charaktere löschen, sowie Raid- und Mythisch+ Informationen ansehen.

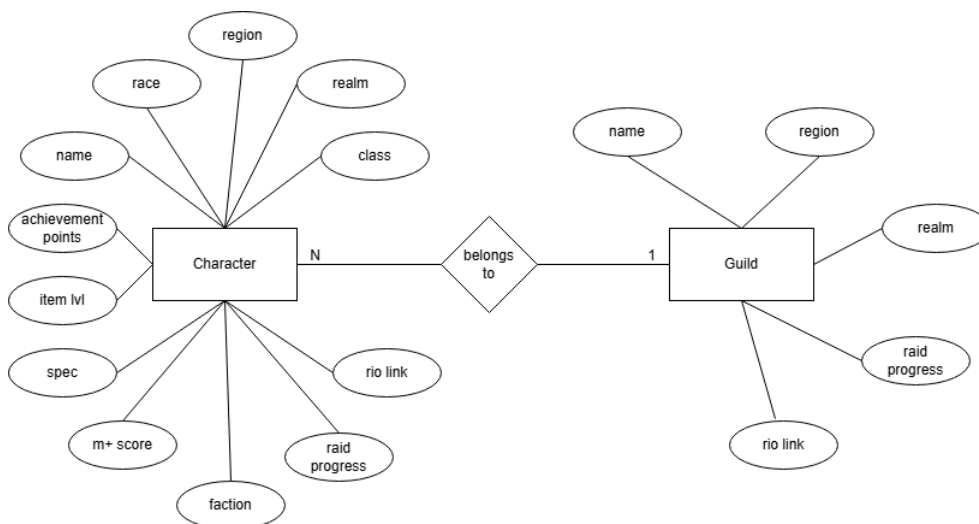
Die Besonderheit: Gilden sind an Charaktere gebunden und werden automatisch beim Import eines Charakters erkannt. Es gibt keine eigenständige Gilden-Übersicht; Gilden-Details sind jedoch über jeden Charakter mit Gilde erreichbar.

Die Anwendung ist mit SvelteKit und MongoDB umgesetzt.

2. Datenmodell

Das Datenmodell besteht aus zwei Entitätstypen:

Character und Guild. Jeder Charakter ist einer Gilde zugeordnet. Die Daten werden in zwei MongoDB Collections gespeichert: characters und guilds



3. Beschreibung der Anwendung

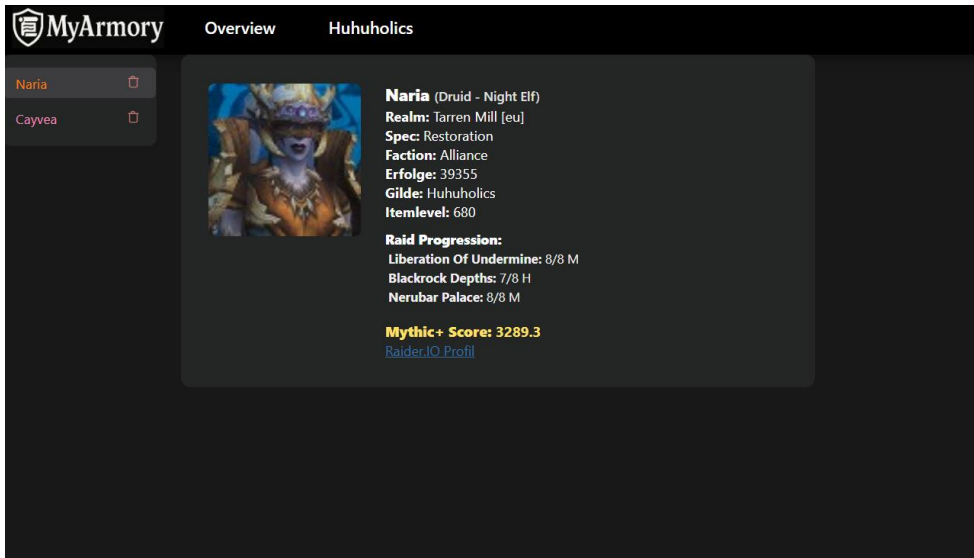
3.1. Charakterübersicht

Route: /overview

Auf dieser Seite werden alle gespeicherten Charaktere angezeigt. Der Nutzer kann einen Charakter auswählen, um Details zu sehen oder ihn zu löschen. Das Hinzufügen neuer Charaktere erfolgt über das Formular auf der Startseite, welches über das MyArmory-Logo oben links zu erreichen ist.

Dateien:

- src/routes/overview/+page.svelte
- src/routes/api/chars/+server.ts
- src/routes/+layout.svelte



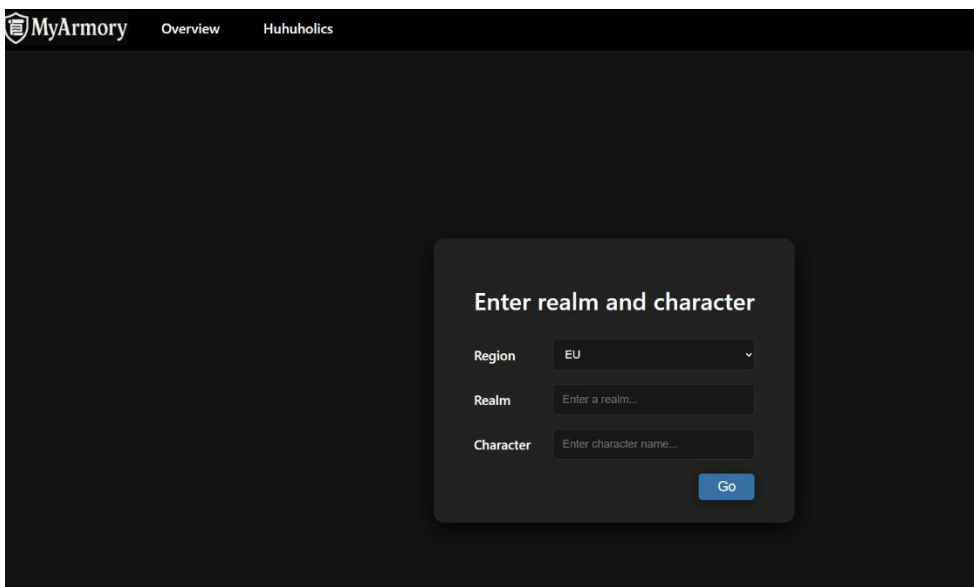
3.2. Character hinzufügen

Route: /

Auf der Startseite befindet sich das Import-Formular, um einen neuen Charakter via Realm und Namen hinzuzufügen. Die Daten werden per Raider.IO API geladen und direkt in die MongoDB geschrieben.

Dateien:

- lib/components/ImportForm.svelte
- lib/components/RealmSelect.svelte
- src/server/db.ts
- src/routes/api/add-character/+server.ts
- src/routes/+layout.svelte
- src/routes/+page.svelte



3.3. Gilden-Detailseite

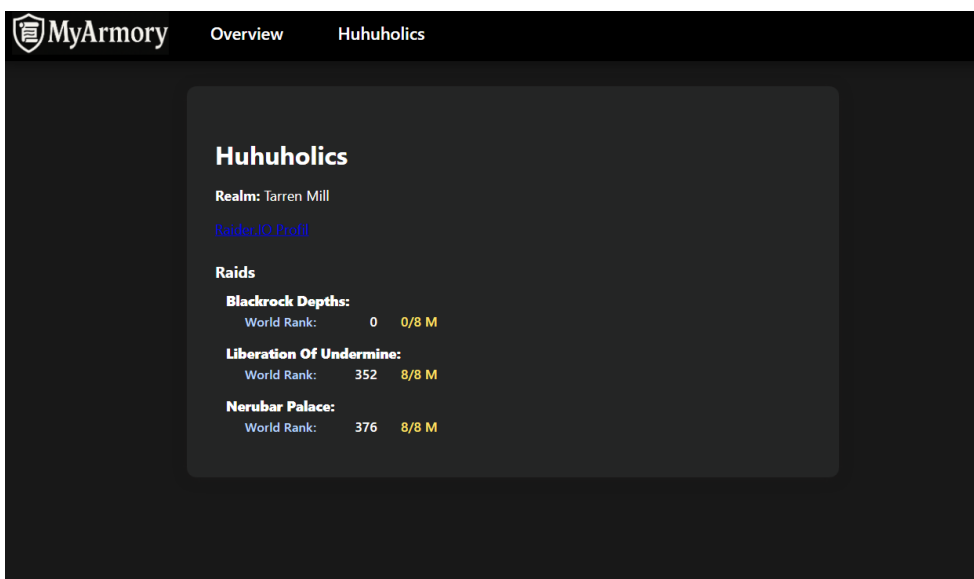
Route: `/guilds/[name]`

Die Gildendetails sind über die Headleiste erreichbar: Klick auf den Gildennamen öffnet die Gilden-Detailseite mit Raidfortschritt, Realm und Raider.io Link.

Hinweis: Eine eigenständige Gilden-Übersicht ist nicht vorgesehen; Gilden werden stets über die Charaktere gefunden.

Dateien:

- `src/routes/guild/+page.svelte`
- `src/routes/api/guild/[region]/[realm]/[name]/+server.ts`
- `src/routes/+layout.svelte`

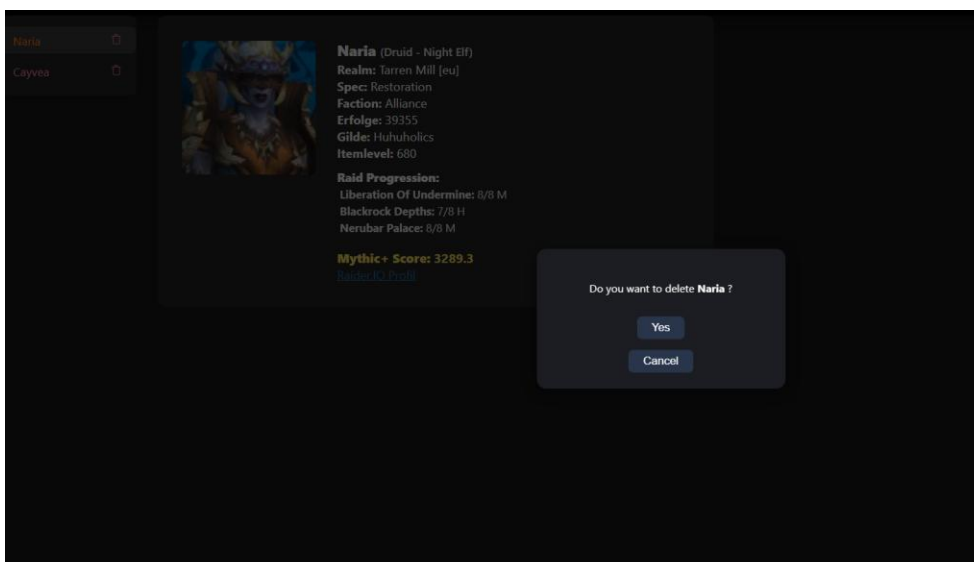


3.4. Charakter löschen

Charaktere können direkt im Overview gelöscht werden. Das Löschen wird sofort im Frontend reflektiert und der Eintrag wird auch dauerhaft in MongoDB entfernt

Dateien:

- `src/api/chars/[region]/[realm]/[name]/+server.ts`



4. Erweiterungen

4.1. Raider.IO API-Integration (automatischer Import der Char- und Gildendaten)

Beschreibung:

Beim Hinzufügen eines neuen Charakters wird nicht die lokale Datenbank verwendet, sondern der Charakter und alle relevanten Informationen werden automatisch über die Raider.IO API geladen. Dadurch sind Raid Fortschritt, M+ Score und Gildeninformationen immer aktuell und es müssen keine statischen Demodaten gepflegt werden.

Code:

- Lib/components/ImportForm.svelte
- src/routes/api/guild/[region]/[realm]/[name]/+server.ts
- src/routes/api/add-character/+server.ts

4.2. Detaillierte Raid- und M+ Fortschrittsanzeige

Beschreibung:

Die App zeigt für jeden Charakter den aktuellen Fortschritt in allen wichtigen Raids sowie den aktuellen M+ Score an. Für Raids wird der höchste Schwierigkeitsgrad dargestellt. Das bietet einen umfassenden Überblick über die Leistung des Charakters.

Code:

- src/routes/api/add-character/+server.ts
- src/routes/overview/+page.svelte

4.3. Automatisches Verknüpfen von Char und Gilde

Beschreibung:

Die Anwendung verknüpft jeden Charakter direkt mit seiner zugehörigen Gilde. Gildeninformationen werden ebenfalls aus Raider.IO geladen und beim Import eines Charakters automatisch in der Datenbank abgelegt bzw. aktualisiert. Dadurch ist eine zentrale Gildenverwaltung möglich, ohne dass der Nutzer diese separat anlegen muss.

Code:

- routes/api/chars/[region]/[realm]/[name]/+server.ts
- Lib/components/ImportForm.svelte
- src/routes/api/add-character/+server.ts
- src/routes/guild/+page.svelte

4.4. Custom-UI mit WoW-Farbschema

Beschreibung:

Die Anwendung nutzt die für World of Warcraft typischen Klassenfarben zur farblichen Darstellung von Namen und Elementen. Das macht das UI übersichtlich und nah an der WoW-Ästhetik.

Code:

- src/routes/overview/+page.svelte

4.5. Dynamisches Löschen von Chars mit Sofort-Update im UI

Beschreibung:

Charaktere können direkt im UI gelöscht werden. Das Löschen wird sofort im Frontend reflektiert und der Eintrag wird auch dauerhaft in MongoDB entfernt. Wenn kein Charakter mehr vorhanden ist, wird automatisch zur Startseite navigiert.

Code:

- `src/api/chars/[region]/[realm]/[name]/+server.ts`
- `src/routes/overview/+page.svelte`

4.6. Gilden-Detailansicht an Char gebunden

Beschreibung:

Die Detailseite einer Gilde zeigt den vollständigen Raidfortschritt der gesamten Gilde, sofern vorhanden. Sie ist direkt über den Link im Char-Detail erreichbar und bietet eine Übersicht über Realm, Raidfortschritt und weitere relevante Daten.

Code:

- `routes/guilds/[name]/+page.svelte`
- `lib/components/GuildDetails.svelte`

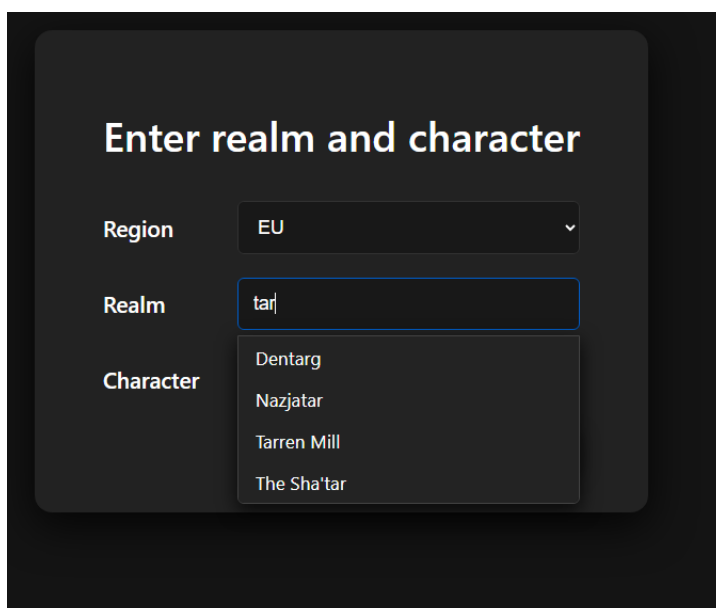
4.7. Realm-Autocompletion/Usability-Feature

Beschreibung:

Beim Hinzufügen eines Characters werden Realmnamen automatisch vervollständigt bzw. die Auswahl aus dem Dropdown-Menü auf die Eingabe angepasst.

Code:

- `lib/components/RealmSelect.svelte`



The screenshot shows a dark-themed form titled "Enter realm and character". It contains three input fields: "Region" with a dropdown menu showing "EU", "Realm" with a text input containing "tarj", and "Character" with a dropdown menu showing a list of character names: "Dentarg", "Nazjatar", "Tarren Mill", and "The Sha'tar".

4.8. Navigation/Usability-Feature

Die Headbar dient als zentrale Navigation der Anwendung. Das Logo führt zur Startseite. Overview zeigt alle Charaktere – und sofern für den gewählten Character vorhanden – wird ein Gildenlink angezeigt, der direkt zur Gildenseite führt. Ist der Character in einer Gilde wird der Reiter Gilde direkt zum Gildennamen.

Code:

- `src/routes/+layout.svelte`
- `static/images/MyArmoryLogo.png`

