

GIBZ FIEŠTA

Projekt Erläuterung

Inhaltlich geht es in der Projektarbeit des Moduls 326 um die Erstellung einer Software, welche von den Lehrpersonen des GIBZ zur langfristigen Evaluation ihrer digitalen, unterrichtsbezogenen Anwendungskompetenzen genutzt werden kann: Aus einem Set definierter Kompetenzen sollen Lehrpersonen per Selbstdeklaration die individuelle Erreichung unterschiedlicher Kompetenzen erfassen und planen. Zur Unterstützung der Lehrpersonen können für die Kompetenzen unterschiedliche Ressourcen bereitgestellt werden. Wir haben uns hier für ein Webtool entschieden. Die Anforderungen an dieses Tool sind unten aufgelistet.

Anforderungen

Funktional

- Rollensystem ("Sys"Admin, Guest, User)
 - Berechtigungen "User":
 - * Kompetenzen erstellen
 - * Kompetenzen löschen
 - * Ressourcen für Kompetenzen zur Verfügung stellen
 - Berechtigungen "Admin":
 - * User löschen, bearbeiten, Account deaktivieren -> CRUD (plus/minus)
 - Berechtigungen "Guest":
 - * Login nicht möglich d.h. Applikation nicht benutzbar -> kann Lehrer - Profile einsehen
- Kompetenzen:
 - "Fields": Name, Skill level (1-4), Links / Hilfestellungen / Material zur Kompetenz (PDF's, Cheatsheets, etc)

Nicht Funktional

Vom PDF

- Muss mit OOP Sprache gemacht werden
- Testing
 - Testkonzept & Testfälle dokumentieren
 - Unit Tests decken >60% ab (API Tests, UI Tests)

Eigene

- Dark-Mode Switch
- Responsive UI Design / Layout
- Blockiert den Loginbutton nach 3 Fehlversuchen
- Erweiterte Logfunktionen:
 - Login / Logout wird Dokumentiert -> Sichtbar für Sysadmin

- Login - IP's werden geloggt
- Auth für API - Requests
- Passwörter hashed (z.B. SHA256)
- Minimaler Traffic -> nur die nötigen Daten schicken

Technologien

Da unser Plan eine Web-Applikation zu entwickeln ist brauchen wir passende Programmiersprachen und Tools, um dieses Vorhaben umzusetzen.

Fürs Frontend haben wir uns für **Svelte** Entschieden. Es ist einfach zu verstehen und vielseitig einsetzbar.

Beim Backend fiel unsere Entscheidung auf **Rust**, da es eine moderne, effiziente bzgl. der Ressourcen und beliebte Sprache ist, die für Backend - Services viel verwendet wird.

Natürlich müssen die Daten auch irgendwo gespeichert werden. Unsere Wahl fällt auf **MongoDB**, da wir mit diesem Tool schon vertraut sind und es Erfahrungsgemäss einfach zu verwenden ist.

Erfahrung

Ramon hat schon kleine Svelte-Kenntnisse, Rust hingegen ist für alle neu. MongoDB allerdings haben wir alle schon genutzt und verstehen wir zum grossen Teil auch recht gut.

Tools

Um die Seite zu entwickeln werden wir verschiedene Tools verwenden. Zum einen wäre da **VSCode**, ein gratis Open-Source Texteditor. Zusätzlich werden wir **Datagrip** und **Webstorm** der JetBrains Toolbox verwenden, um allfällige Probleme mit **VSCode** zu kompensieren.