Projekt-Dokumentation - Lernprogramm - Beleg Internettechnologien I

Lotte Richter
s85482
Matrikelnummer: 53042

27-05-2024

Contents

1 Projektübersicht

Dieses Projekt ist eine Progressive Web App(PWA) für ein Lernprogramm, welches auch offlinefähig ist. offlinefähig.

Das Programm ist ein Lernprogramm, dessen Funktionalität darin besteht, Fragen zu beantworten und damit einen Lernfortschritt zu erreichen. Dazu kann man aus verschiedenen Kategorien wählen aus denen zufällige Fragen, passend zur Kategorie, gewählt werden.

Zu jeder Frage gibt es vier Antworten. Je nachdem, ob man die Frage richtig beantwortet, färbt sich der Button der Antwort rot oder grün ein und die entsprechende Progress-Bar wird erhöht. Ist die Antwort falsch, so kann man sie direkt erneut beantworten. Ist die Antwort richtig, so wird eine neue Frage geladen.

Wenn eine der beiden Progress-Bars 100% erreicht, wird eine kurze Auswertung angezeigt.

2 Verzeichnisstruktur

Das Projekt ist folgendermaßen strukturiert:

```
Lernprogramm
├── doc
│   ├── Projekt-Dokumentation_Richter_Lotte.tex
│   └── Projekt-Dokumentation_Richter_Lotte.pdf
├── images
│   └── demo.png
├── scripts
│   └── sw.js
├── src
│   ├── lernprogramm.css
│   ├── lernprogramm.html
│   ├── lernprogramm.js
│   └── manifest.json
├── Beleg-Aufgabenformat.md
├── Beleg-Aufgabenstellung.md
└── README.md
```

3 Technologien und Abhängigkeiten

- HTML5: Struktur der Webanwendung
- CSS3: Styling der Anwendung
- JavaScript: Funktionalität der Anwendung
- Service Worker: Offline-Fähigkeit und Caching
- Fetch API: Abrufen von Quizfragen von einem externen Server
- PWA Manifest: Definition der PWA-Eigenschaften
- KaTeX: Rendering von mathematischen Formeln

Bei dem Entwurf des Grundgerüsts des Programmes habe ich mich an den Beispielen aus der Vorlesung und Praktikum orientiert.

Um den Quizserver nutzen zu können, muss man sich im Netz der HTW Dresden befinden, da sonst der Server nicht erreicht werden kann, von dem die Fragen geholt werden.

4 Weiterentwicklung

Konfigurationen:

- Es können mehr Fragen vom Quizserver geholt werden, wenn man die Konstante SERVER_QUESTIONS in der Datei lernprogramm.js erhöht. Die Konstante gibt an, wie viele Pages vom Server geholt werden. Dabei besitzt eine Page zehn Fragen.
- Es kann die Anzahl an beantworteten Fragen geänder werden, die es braucht bis eine Progress Bar 100% erreicht. Dazu wird die Konstante TOTAL_QUESTIONS in der Datei lernprogramm.js verwendet.
- Die CSS-Formatierung der Progress Bars funktioniert nur mit dem Browser Firefox. Wenn man eine anderen Browser nutzt, enpfiehlt es sich die im Code markierten Stellen zu ändern.

Erweiterung des Fragenkatalogs:

- Weitere Fragen können in den jeweiligen Kategorien im 'questions' Objekt hinzugefügt werden.

UI-Verbesserungen:

- Man kann die CSS-Datei anpassen, um die Benutzeroberfläche des Lernprogramms zu verbessern.

Button Highlight der Kategorie Quizserver

- Nach dem Anklicken eines Buttons wird dieser rot oder grün hervorgehoben, je nachdem ob die Antwort richtig ist. Da beim Quizserver die Antworten anders geprüft werden, als bei den lokalen Fragen und das ganze asynchron läuft, ist mir keine elegante Lösung zur Hervorhebung der Buttons gefunden.

5 Nutzung von KI

5.1 ChatGPT

Zum Entwicklen des Programmes wurd an manchen Stellen die KI "ChatGPT" genutzt. Hauptsächlich zu Beginn des Projektes, sobald das Projekt und die Quellcodes ein bisschen umfangreicher wurde, war die KI nicht mehr so hilfreich.

Genutzt wurde ChatGPT bei simplen Aufgaben, wie das Einlesen der Daten aus der JSON-Datei, oder bei der Erstellung von Dokumenten, wie. z.B. die Projekt-Dokumentation.