

Radio Player

Übersicht

Der Radio Player ist eine Webanwendung, die es Benutzern ermöglicht, ihre Lieblingsradiosender online zu hören. Die Anwendung bietet Funktionen wie die Auswahl von Radiosendern nach Land und Genre, eine Suchfunktion und die Möglichkeit, zwischen einem Hell- und Dunkelmodus zu wechseln.

Anleitung zur Nutzung des Radio Players

1. ****Startseite aufrufen****: Öffnen Sie die `index.html` -Datei in einem Webbrowser oder den folgenden Link: <https://altendeh.github.io/WebPortfolio/index.html> .
2. ****Radiosender filtern****: Wählen Sie ein Land und/oder ein Genre aus den Dropdown-Menüs aus oder geben Sie den Namen eines Radiosenders in das Suchfeld ein.
3. ****Radiosender suchen****: Klicken Sie auf den Suchbutton oder drücken Sie die Eingabetaste, um die Suche zu starten.
4. ****Radiosender auswählen****: Klicken Sie auf einen Radiosender in der Liste, um zur Detailseite zu gelangen.
5. ****Radiosender abspielen****: Klicken Sie auf den Play-Button, um die Wiedergabe zu starten. Verwenden Sie den Lautstärkeregler, um die Lautstärke anzupassen.
6. ****Modus wechseln****: Klicken Sie auf den Modus-Button im Header, um zwischen Hell- und Dunkelmodus zu wechseln.
7. ****Zurück zur Senderauswahl****: Klicken Sie auf den Zurück-Button, um zur Hauptseite zurückzukehren und einen anderen Sender auszuwählen.

****Viel Spaß beim Hören Ihrer Lieblingsradiosender!****

Vorgehen

1. Planung und Anforderungen

Zunächst wurde überlegt, welche Funktionalität die Web-Anwendung abbilden soll. Nach der Entscheidung für einen Radio Player wurden die Anforderungen in einem Brainstorming formuliert und andere Radio- und Musikanwendungen verglichen. Daraus ergaben sich die folgenden Funktionen und der Aufbau der Anwendung.

2. Aufbau der Anwendung

Die Anwendung ist in zwei Seiten aufgeteilt:

- ****Seite 1: Radiosender suchen und filtern****
 - Möglichkeit, nach Radiosendern zu suchen
 - Filter nach Genre und Land
 - Ausgabe einer Liste mit den Top 10 Ergebnissen

- ****Seite 2: Radiosender abspielen****
 - Lautstärke anpassen
 - Abspielen pausieren
 - Informationen über den Radiosender (Land, Genre, Icon/Bild)
 - Navigation zurück zur ersten Seite

3. Implementierung

Die Implementierung erfolgte in mehreren Schritten:

1. ****Hauptmenü und Suchfunktionen****

- Programmierung des Abrufs der Radiosender in JavaScript
- Implementierung der Filter- und Suchfunktionen

2. ****Seitenaufbau und Design****

- Anpassung des Seitenaufbaus für eine ansprechendere Benutzeroberfläche

- Implementierung der Funktionen der zweiten Seite und Anpassung des Designs

3. ****Zusätzliche Funktionen****

- Einfügen eines Dunkel-/Hellmodus
- Nutzung von Session Storage, um das erneute Laden der Radiosender nach jeder Rückkehr auf die Hauptseite zu vermeiden

4. ****Responsive Design****

- Hinzufügen von Media Queries, um die Anwendung für Geräte mit kleineren Bildschirmen responsive zu gestalten

4. Dokumentation

Die Dokumentation im Code und im ReadMe wurde während und nach der Arbeit ausgearbeitet.

Herausforderung

Implementieren des Hell- Dunkelmodus

Bei der Implementierung des Hell-Dunkelmodus lag das Problem vor, dass nicht die passenden Farben bzw. die passenden Icons angezeigt wurden. Wenn der Dunkelmodus aktiviert war, war das Design hell und das Zeichen des Buttons zum Umschalten des Modus falsch. Zudem funktioniert die ursprünglich festgelegten und erstellten Designs nicht. Dabei wurde zum Beispiel das Icon des Radiosenders nicht mehr angezeigt.

Fehler bei der Filteroption

Beim Filtern nach einem Genre lag zunächst das Problem vor, dass egal welches Genre ausgewählt wurde, kein Radiosender angezeigt wurde. Das Problem lag daran, dass das Genre des Radiosenders nicht über den Begriff Genre definiert war, sondern über den Begriff Tag. Nach dem Anpassen der Schnittstelle funktionierte die Filterfunktion für die Genres.

Anpassen des responsive Designs:

Die Webanwendung wurde zunächst für Desktops konzipiert. Da die Anwendung auch auf kleineren Bildschirmen genutzt werden soll, muss das Design auch für diese Bildschirmgrößen angepasst ist und die alle Funktionalitäten auch für diese Geräte gewährleistet sind. In diesem Projekt wurde das Webdesignkonzept "Mobile-First Design" nicht befolgt, dadurch erhöht sich die Komplexität, die für die Gestaltung und Anpassung des Designs von Desktopgröße zur Größe von Bildschirmen mobiler Geräte.

Steuerung der Lautstärke über Lautstärketasten

Eine weitere Überlegung war die Implementierung einer Synchronisation der Lautstärke der Anwendung mit der Lautstärke des Gerätes. Im weiteren sollte die Lautstärke dabei über die Hardware-Lautstärketasten gesteuert werden. Eine direkte Umsetzung von diesem war nicht möglich, da die Synchronisation dieser nicht aufgebaut werden konnte. Die Lautstärkereglung wird daher über einen Lautstärkeregler in der Webanwendung realisiert.

Dateien und Struktur

index.html

Diese Datei enthält die Hauptstruktur der Webseite. Sie bindet die notwendigen CSS- und JavaScript-Dateien ein und definiert die grundlegenden Elemente der Benutzeroberfläche, wie den Header, die Filteroptionen, die Ladeanzeige und die Liste der Radiosender.

- **Header**: Enthält den Titel des Radio Players und einen Button zum Umschalten zwischen Hell- und Dunkelmodus.
- **Filteroptionen**: Ermöglichen die Auswahl von Radiosendern nach Land und Genre sowie die Suche nach einem spezifischen Sender.
- **Ladeanzeige**: Wird angezeigt, während die Radiosender geladen werden.
- **Liste der Radiosender**: Zeigt die verfügbaren Radiosender basierend auf den ausgewählten Filtern und Suchbegriffen an.

radio.html

Diese Datei zeigt die Detailseite eines ausgewählten Radiosenders an. Sie enthält die Elemente zur Anzeige der Senderinformationen und zur Steuerung der Audiowiedergabe.

- **Senderinformationen**: Zeigt den Namen, das Icon, das Land und das Genre des ausgewählten Radiosenders an.
- **Audiosteuerung**: Ermöglicht das Abspielen und Pausieren des Radios sowie die Anpassung der Lautstärke.
- **Zurück-Button**: Ermöglicht die Rückkehr zur Hauptseite mit der Senderauswahl.

style.css

Diese Datei enthält die CSS-Stile für die gesamte Anwendung. Sie definiert das Layout, die Farben und die Animationen für die verschiedenen Elemente der Benutzeroberfläche.

- **Farbvariablen**: Definiert Farbvariablen für den Hell- und Dunkelmodus.
- **Grundlegende Stile**: Enthält grundlegende Stile für den Body, den Header, den Hauptbereich und die Fußzeile.
- **Filteroptionen**: Stile für die Dropdowns und das Suchfeld, einschließlich Hover-Effekte.
- **Audiosteuerung**: Stile für die Steuerungsknöpfe und den Lautstärkeregler.
- **Media Queries**: Anpassungen für mobile Geräte, um das Layout entsprechend zu ändern.

script.js

Diese Datei enthält die JavaScript-Funktionalität für die Hauptseite der Anwendung. Sie steuert das Abrufen und Anzeigen der Radiosender sowie den Moduswechsel.

- **DOM-Elemente abrufen**: Holt die benötigten Elemente aus dem DOM.
- **Event-Listener hinzufügen**: Fügt Event-Listener für den Suchbutton, die Eingabetaste im Suchfeld und das Laden des Dokuments hinzu.
- **fetchRadios**: Funktion zum Abrufen der Radiosender von der API, basierend auf den ausgewählten Filtern und Suchbegriffen.
- **displayRadios**: Funktion zum Anzeigen der abgerufenen Radiosender im DOM.
- **Moduswechsel**: Funktionen zum Umschalten zwischen Hell- und Dunkelmodus und zum Aktualisieren des Modus-Icons.

radio.js

Diese Datei enthält die JavaScript-Funktionalität für die Detailseite eines ausgewählten Radiosenders. Sie steuert die Anzeige der Senderinformationen und die Audiowiedergabe.

- **DOMContentLoaded**: Wartet, bis das DOM vollständig geladen ist, bevor der Code ausgeführt wird.
- **selectedRadio**: Holt den ausgewählten Radiosender aus dem Session Storage.
- **Elemente abrufen**: Holt die benötigten Elemente aus dem DOM.
- **Überprüfung des ausgewählten Radiosenders**: Überprüft, ob ein Radiosender ausgewählt wurde, und zeigt dessen Informationen an.
- **Play-Button**: Fügt einen Event-Listener hinzu, um die Wiedergabe des Radios zu starten oder zu pausieren.
- **Lautstärkeregler**: Fügt einen Event-Listener hinzu, um die Lautstärke des Radios zu steuern.
- **Lautstärke-Icon**: Fügt einen Event-Listener hinzu, um die Lautstärke stummzuschalten oder wiederherzustellen.
- **Zurück-Button**: Fügt einen Event-Listener hinzu, um zur Hauptseite zurückzukehren.
- **Modus-Button**: Fügt einen Event-Listener hinzu, um zwischen Hell- und Dunkelmodus zu wechseln.

- ****setMode****: Funktion zum Setzen des Modus (Hell oder Dunkel).
- ****updateModelcon****: Funktion zum Aktualisieren des Modus-Icons.

Wichtige Funktionen

Hell- und Dunkelmodus

Die Anwendung bietet die Möglichkeit, zwischen einem Hell- und Dunkelmodus zu wechseln. Dies wird durch den Button im Header gesteuert. Der aktuelle Modus wird im Session Storage gespeichert, sodass er beim nächsten Laden der Seite beibehalten wird.

Filteroptionen

Benutzer können Radiosender nach Land und Genre filtern. Zusätzlich gibt es ein Suchfeld, um nach spezifischen Sendern zu suchen. Die Filteroptionen und die Suche werden durch Dropdown-Menüs und ein Eingabefeld realisiert.

Audiosteuerung

Auf der Detailseite eines ausgewählten Radiosenders können Benutzer die Wiedergabe starten oder pausieren und die Lautstärke anpassen. Es gibt auch ein Lautstärke-Icon, das die Lautstärke stumm schalten oder wiederherstellen kann.

Technische Hintergründe

Nutzung von Session Storage

****Session Storage**** ist eine Web-API, die es ermöglicht, Daten für die Dauer einer Sitzung im Browser zu speichern. Diese Daten bleiben erhalten, solange der Browser geöffnet ist und werden gelöscht, sobald der Browser oder der Tab geschlossen wird. In

der Radio Player-Anwendung wird Session Storage verwendet, um den aktuellen Modus (Hell oder Dunkel) und die ausgewählten Radiosender zu speichern.

Vorteile von Session Storage:

- **Einfach zu verwenden**: Session Storage bietet eine einfache API zum Speichern und Abrufen von Daten.
- **Sitzungsbasiert**: Daten bleiben nur für die Dauer der Sitzung erhalten, was für temporäre Daten nützlich ist.
- **Sicher**: Daten sind nur im aktuellen Tab oder Fenster verfügbar und können nicht von anderen Tabs oder Fenstern ausgelesen werden.

Beispiel in der Anwendung:

- **Speichern des Modus**: Der aktuelle Modus (Hell oder Dunkel) wird im Session Storage gespeichert, damit er beim nächsten Laden der Seite beibehalten wird.
- **Speichern des ausgewählten Radiosenders**: Der ausgewählte Radiosender wird im Session Storage gespeichert, um die Detailseite mit den entsprechenden Informationen zu laden.

Abrufen und Anzeigen von Radiosendern

Die Anwendung verwendet die Fetch API, um Radiosender von einer externen API abzurufen. Die abgerufenen Daten werden dann gefiltert und im DOM angezeigt.

Schritte:

1. **Fetch API**: Die Fetch API wird verwendet, um eine Anfrage an die Radio-Browser-API zu senden und die Radiosender abzurufen.
2. **Filtern der Daten**: Die abgerufenen Daten werden basierend auf den ausgewählten Filtern (Land, Genre, Suchbegriff) gefiltert.
3. **Anzeigen der Daten**: Die gefilterten Radiosender werden im DOM angezeigt, indem neue Elemente erstellt und eingefügt werden.

Moduswechsel (Hell- und Dunkelmodus)

Die Anwendung bietet die Möglichkeit, zwischen einem Hell- und Dunkelmodus zu wechseln. Dies wird durch das Hinzufügen oder Entfernen von CSS-Klassen gesteuert.

Schritte:

1. **Event-Listener**: Ein Event-Listener wird dem Modus-Button hinzugefügt, um auf Klicks zu reagieren.
2. **CSS-Klassen**: Basierend auf dem aktuellen Modus werden die entsprechenden CSS-Klassen (`light-mode` oder `dark-mode`) zum Body-Element hinzugefügt oder entfernt.
3. **Speichern im Session Storage**: Der aktuelle Modus wird im Session Storage gespeichert, damit er beim nächsten Laden der Seite beibehalten wird.

Audiosteuerung

Die Audiosteuerung ermöglicht das Abspielen, Pausieren und Anpassen der Lautstärke des Radios.

Schritte:

1. **Audio-Element**: Ein neues Audio-Element wird erstellt und die URL des ausgewählten Radiosenders wird als Quelle gesetzt.
2. **Play/Pause**: Ein Event-Listener wird dem Play-Button hinzugefügt, um die Wiedergabe zu starten oder zu pausieren.
3. **Lautstärkeregler**: Ein Event-Listener wird dem Lautstärkeregler hinzugefügt, um die Lautstärke des Audio-Elements anzupassen.
4. **Lautstärke-Icon**: Ein Event-Listener wird dem Lautstärke-Icon hinzugefügt, um die Lautstärke stummzuschalten oder wiederherzustellen.

API-Details

Die Radio Player-Anwendung nutzt die Radio Browser API, um Radiosender abzurufen. Diese API bietet eine Vielzahl von Endpunkten, um Radiosender nach verschiedenen Kriterien zu suchen und zu filtern.

Wichtige Endpunkte

- **Top bewertete Sender:**

- **URL:** `https://de1.api.radio-browser.info/json/stations/topvote`

- **Beschreibung:** Ruft die am besten bewerteten Radiosender ab.

- **Sender nach Name suchen:**

- **URL:** `https://de1.api.radio-browser.info/json/stations/search?name={searchTerm}`

- **Beschreibung:** Sucht nach Radiosendern, deren Name den Suchbegriff enthält.

- **Beispiel:** `https://de1.api.radio-browser.info/json/stations/search?name=rock`

- **Sender nach Land filtern:**

- **URL:** `https://de1.api.radio-browser.info/json/stations/bycountry/{country}`

- **Beschreibung:** Ruft Radiosender aus einem bestimmten Land ab.

- **Beispiel:** `https://de1.api.radio-browser.info/json/stations/bycountry/Germany`

- **Sender nach Genre filtern:**

- **URL:** `https://de1.api.radio-browser.info/json/stations/bytag/{tag}`

- **Beschreibung:** Ruft Radiosender eines bestimmten Genres ab.

- **Beispiel:** `https://de1.api.radio-browser.info/json/stations/bytag/pop`

- **Sender nach Land und Genre filtern:**

- **URL:** `https://de1.api.radio-browser.info/json/stations/search?country={country}&tag={tag}`

- **Beschreibung:** Sucht nach Radiosendern, die sowohl das Land als auch das Genre erfüllen.

- **Beispiel:** `https://de1.api.radio-browser.info/json/stations/search?country=Germany&tag=rock`

Font Awesome Bibliothek

Font Awesome ist eine beliebte Bibliothek, die eine Vielzahl von Icons zur Verfügung stellt, die in Webprojekten verwendet werden können. Diese Icons sind skalierbar und können leicht angepasst werden, um den Stil und das Design der Webseite zu ergänzen.

Einbindung in das Projekt

In den HTML-Dokumenten wird Font Awesome über ein CDN (Content Delivery Network) eingebunden. Dies geschieht durch das Hinzufügen des folgenden `<link>`-Tags im `<head>`-Bereich des Dokuments:

```
```html<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.3/css/all.min.css">```
```

#### Verwendung von Icons

Nach der Einbindung der Bibliothek können Icons in deinem HTML-Dokument verwendet werden, indem du die entsprechenden Klassen hinzufügst. Zum Beispiel:

```
```html
```

```
<i class="fas fa-radio"></i>
```

```
` ``
```

In diesem Beispiel wird das Radio-Icon von Font Awesome verwendet. Die Klasse `fas` steht für "Font Awesome Solid", und `fa-radio` ist der spezifische Name des Icons.

Vorteile der Verwendung von Font Awesome:

- **Vielfalt an Icons**: Font Awesome bietet eine große Auswahl an Icons für verschiedene Anwendungsfälle.
- **Skalierbarkeit**: Icons sind vektorbasiert und können ohne Qualitätsverlust skaliert werden.
- **Einfache Integration**: Icons können einfach durch Hinzufügen von Klassen in HTML-Elemente integriert werden.
- **Anpassbarkeit**: Icons können durch CSS leicht angepasst werden, um Farbe, Größe und andere Stileigenschaften zu ändern.

Beispiel

Hier ist ein Beispiel, wie ein Icon in einem Button verwendet wird:

```
` `` html  
  
<button>  
  <i class="fas fa-adjust"></i> Umschalten  
</button>  
` ``
```

In diesem Beispiel wird das ``fa-adjust`` Icon von Font Awesome verwendet, um einen Button zu erstellen, der zwischen Hell- und Dunkelmodus umschaltet.