

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Домашняя работа № 2

Выполнила:

Иржанова Юлия

Группа

К3344

Проверил:

Добряков Д. И.

Санкт-Петербург
2025 г.

Задача:

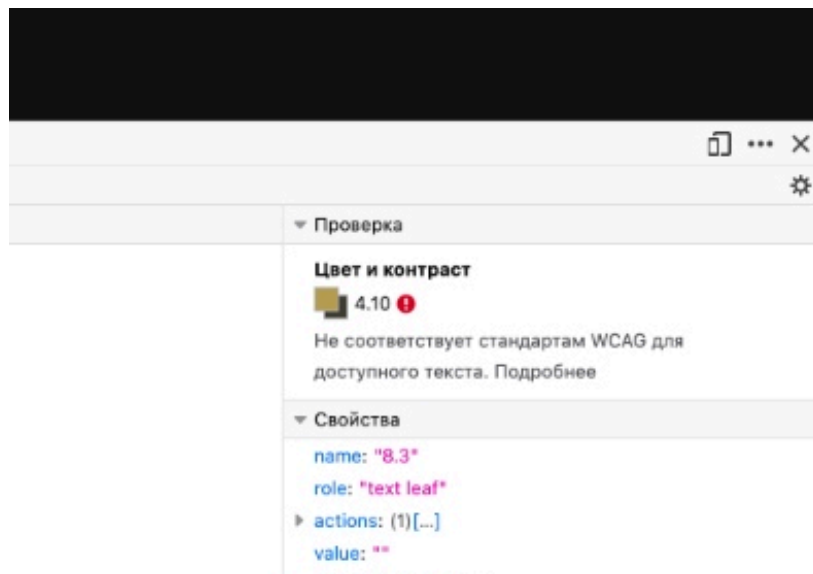
улучшить доступность ранее реализованного сайта. Добавить необходимые HTML-атрибуты ко всему контенту на странице и проверить это с помощью инструментов из Dev Tools браузера Firefox и сервиса Google Lighthouse

Ход работы

Сначала я использовала браузер Mozilla Firefox, с помощью DevTools были выявлены следующие проблемы и ограничения моего сайта и также решения к ним:

- *Страница каталога*

Проблема с контрастом:



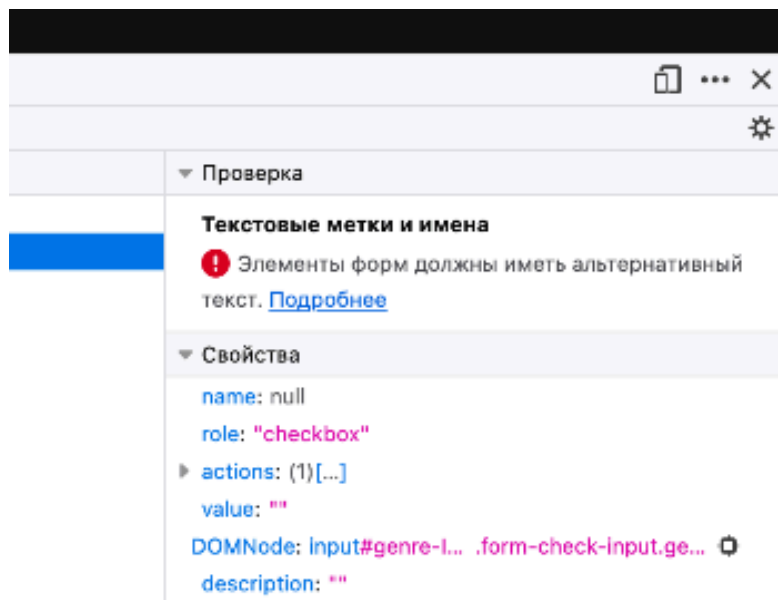
Решение:
style.css

```

}
.album-rating {
  position: absolute;
  right: 18px;
  top: 10px;
  background: #ffc107;
  color: #000000;
  font-weight: 600;
  font-size: 17px;
  border-radius: 8px;
  padding: 7px 13px 6px 13px;
}

```

Проблема “Клавиатура”:



Изначально у инпута есть был только `id` и `value`. Добавила `name` и `aria-label`, чтобы даже без `label` у чекбокса было собственное имя:

```

function renderGenreFilters() {
  const genreFilters = document.getElementById('genreFilters');
  if (!genreFilters) return;

  const genres = [...new Set(MOCK_ALBUMS.map(album => album.genre))];

  genreFilters.innerHTML = '';
  genres.forEach(genre => {
    const wrapper = document.createElement('div');
    wrapper.className = 'form-check';

    const input = document.createElement('input');
    input.type = 'checkbox';
    input.className = 'form-check-input genre-checkbox';
    input.id = `genre-${genre}`;
    input.name = 'genre';
    input.value = genre;
    input.setAttribute('aria-label', genre);

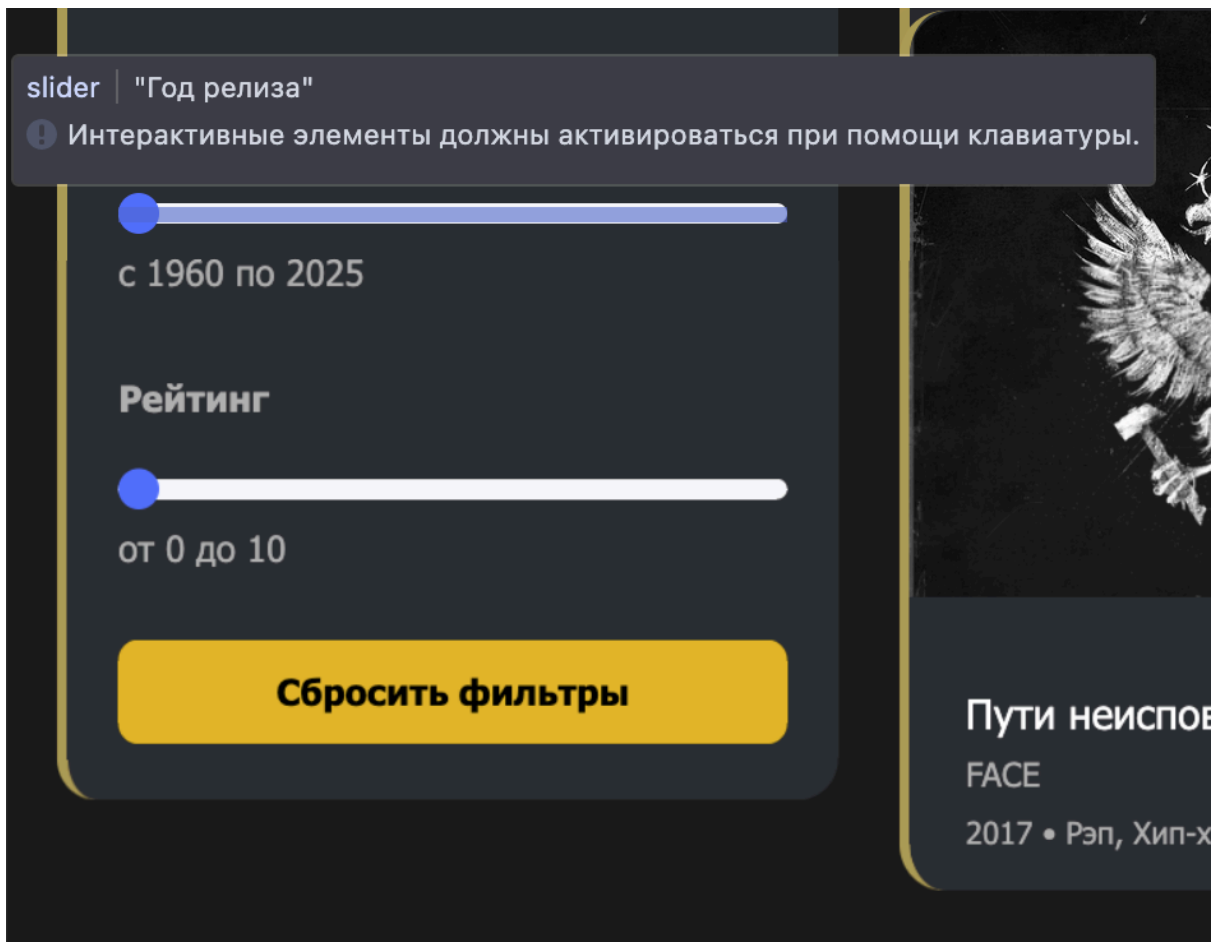
    const label = document.createElement('label');
    label.className = 'form-check-label';
    label.htmlFor = input.id;
    label.textContent = genre;

    wrapper.appendChild(input);
    wrapper.appendChild(label);
    genreFilters.appendChild(wrapper);
  });
}

```

Для чекбоксов жанров добавлены связанные `<label>` и `aria-labelledby`, чтобы обеспечить альтернативный текст и соответствие рекомендациям WCAG

Проблема:



Для слайдеров добавлены label и ARIA-атрибуты, гарантирующие управление с клавиатуры и корректное имя элемента

```
<label for="yearFilter">Год релиза</label>
```

```
<input
```

```
  id="yearFilter"
```

```
  type="range"
```

```
  min="1960"
```

```
  max="2025"
```

```
  value="1960"
```

```
  aria-label="Год релиза"
```

```
  aria-valuemin="1960"
```

```
  aria-valuemax="2025"
```

```
<span id="yearValue">1960</span>
```

Google Lighthouse



Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

ADDITIONAL ITEMS TO MANUALLY CHECK (10)	Hide
<input type="radio"/> Interactive controls are keyboard focusable	▼
<input type="radio"/> Interactive elements indicate their purpose and state	▼
<input type="radio"/> The page has a logical tab order	▼
<input type="radio"/> Visual order on the page follows DOM order	▼
<input type="radio"/> User focus is not accidentally trapped in a region	▼
<input type="radio"/> The user's focus is directed to new content added to the page	▼
<input type="radio"/> HTML5 landmark elements are used to improve navigation	▼
<input type="radio"/> Offscreen content is hidden from assistive technology	▼
<input type="radio"/> Custom controls have associated labels	▼
<input type="radio"/> Custom controls have ARIA roles	▼

These items address areas which an automated testing tool cannot cover. Learn more in our guide on [conducting an accessibility review](#).

Автоматические проверки Lighthouse не выявили ошибок на проверяемой странице.

Аналогично я увеличила доступность на каждой странице. Для удобства я пользовалась только сервисом Google Lighthouse

- *Страница альбома*

Было:

96

Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

CONTRAST

▲ Background and foreground colors do not have a sufficient contrast ratio.

Low-contrast text is difficult or impossible for many users to read. [Learn how to provide sufficient color contrast.](#)

Failing Elements

ОТЗЫВ

Добавить в избранное

button#favoriteBtn.btn.btn-outline-danger.btn-lg

body

These are opportunities to improve the legibility of your content.

Исправила:

```
1 }
2
3 .btn-outline-danger {
4   color: #ff6b6b; /* более яркий красный вместо var(--danger-color) */
5   border: 2px solid #ff6b6b;
6   background-color: transparent;
7   font-weight: 600;
8 }
9
10
11 .btn-outline-danger:hover {
12   background-color: #ff6b6b;
13   color: #ffffff; /* белый текст на красном фоне */
14 }
```

Стало:

100

Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

ADDITIONAL ITEMS TO MANUALLY CHECK (40)

100%

- Страница регистрации

Было:

95

Accessibility

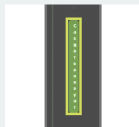
These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

CONTRAST

▲ Background and foreground colors do not have a sufficient contrast ratio.

Low-contrast text is difficult or impossible for many users to read. [Learn how to provide sufficient color contrast.](#)


Failing Elements



button.btn.btn-success.w-100.mb-3




p.mb-0



section.card.shadow-lg



a.text-decoration-none



section.card.shadow-lg

Исправила:

```
.btn-success {
  background-color: #28a745;
  border-color: #28a745;
  color: #ffffff; /* белый текст вместо стандартного */
  font-weight: 600;
}

.btn-success:hover {
  background-color: #218838;
  border-color: #218838;
  color: #ffffff;
}
```



```

/* Ссылка в тексте регистрации */
a.text-decoration-none {
  color: #1DB954; /* яркий зелёный */
  font-weight: 600;
  text-decoration: none;
}

a.text-decoration-none:hover {
  color: #1ed760; /* ещё ярче при наведении */
  text-decoration: underline;
}

```

```

.card {
  background-color: var(--spotify-card);
  border: 1px solid var(--spotify-border);
  color: var(--text-primary); /* белый текст */
}

```

- Страница входа



Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

ADDITIONAL ITEMS TO MANUALLY CHECK (10)

Hide

Сразу все было отлично



- Страница профиля



Accessibility

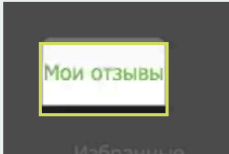
These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

CONTRAST

 Background and foreground colors do not have a sufficient contrast ratio. 

Low-contrast text is difficult or impossible for many users to read. [Learn how to provide sufficient color contrast.](#)

Failing Elements



button#reviewsTab.nav-link.active

```
.nav-link.active {  
  color: #054a1d !important;  
  background-color: transparent;  
  border-bottom-color: #1DB954;  
  font-weight: 600;  
}
```



Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.