

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Фронт-энд разработка

Отчет

Лабораторная работа №1

Вёрстка сайта средствами HTML, CSS и Bootstrap

Выполнил:

Зайцев Кирилл

K33402

Проверил:

Добряков Д. И.

Санкт-Петербург

2023 г.

Задача

Выбранный вариант: Музыкальный плеер.

Задача: Создание веб-приложения для воспроизведения музыки, включающего функции поиска и добавления треков, создания плейлистов и контроля громкости.

Ход работы

Файл registration.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Регистрация</title>
  <style>
    body {
      margin: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      background-color: #f0f0f0;
      font-family: Arial, sans-serif;
    }
    .form-container {
      text-align: center;
      padding: 20px;
      border-radius: 10px;
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
      width: 300px;
      padding: 10px;
      margin: 10px 0;
      border-radius: 5px;
      border: 1px solid #ccc;
      font-size: 16px;
    }
    input[type="submit"] {
      background-color: #4CAF50;
      color: white;
      cursor: pointer;
    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
      /* ... (предыдущие стили) ... */
      transition: all 0.3s ease;
    }
    input[type="text"]:focus,
    input[type="password"]:focus {
      border-color: #4CAF50;
      box-shadow: 0 0 5px rgba(76, 175, 80, 0.5);
    }
  </style>
</head>
<body>
  <div class="form-container">
    <input type="text" value="Имя" />
    <input type="password" value="Пароль" />
    <input type="submit" value="Зарегистрироваться" />
  </div>
</body>
</html>
```

```

    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }

    .error-message {
        color: #f00;
        margin-top: 5px;
        display: none;
    }
}
</style>
</head>
<body>
<div class="form-container">
    <h2>Регистрация</h2>
    <form id="registrationForm">
        <label for="username"></label><input type="text" id="username"
placeholder="Имя пользователя">
        <br>
        <label for="email"></label><input type="text" id="email"
placeholder="Email">
        <br>
        <label for="password"></label><input type="password" id="password"
placeholder="Пароль">
        <br>
        <label for="confirmPassword"></label><input type="password"
id="confirmPassword" placeholder="Повторите пароль">
        <br>
        <input type="submit" value="Зарегистрироваться">
    </form>
    <p>Есть аккаунт? <a href="login.html">Войти</a></p>
    <div class="error-message" id="registrationError">Введите корректные
данные</div>
</div>
<script src="js/registration.js"></script>
<script>
    const registrationForm = document.getElementById('registrationForm');
    const usernameInput = document.getElementById('username');
    const emailInput = document.getElementById('email');
    const passwordInput = document.getElementById('password');
    const confirmPasswordInput = document.getElementById('confirmPassword');
    const registrationError = document.getElementById('registrationError');

    registrationForm.addEventListener('submit', function(event) {
        event.preventDefault();
        if (!validateRegistration(usernameInput.value, emailInput.value,
passwordInput.value, confirmPasswordInput.value)) {
            registrationError.style.display = 'block';
        } else {
            // Simulate successful registration
            alert('Успешная регистрация!');
            window.location.href = 'login.html';
        }
    });

    function validateRegistration(username, email, password, confirmPassword)
{
        const emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
        return username.trim() !== '' &&
            emailRegex.test(email) &&
            password.length > 4 &&
            password === confirmPassword;
    }
</script>

```

```
</body>
</html>
```

Файл login.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Вход</title>
  <style>
    body {
      margin: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      background-color: #f0f0f0;
      font-family: Arial, sans-serif;
    }
    .form-container {
      text-align: center;
      padding: 20px;
      border-radius: 10px;
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
      width: 300px;
      padding: 10px;
      margin: 10px 0;
      border-radius: 5px;
      border: 1px solid #ccc;
      font-size: 16px;
    }
    input[type="submit"] {
      background-color: #4CAF50;
      color: white;
      cursor: pointer;
    }
  </style>
</head>
<body>
  <div class="form-container">
    <input type="text" value="Имя" />
    <input type="password" value="Пароль" />
    <input type="submit" value="Вход" />
  </div>
</body>
</html>
```

```

    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
        /* ... (предыдущие стили) ... */
        transition: all 0.3s ease;
    }

    input[type="text"]:focus,
    input[type="password"]:focus {
        border-color: #4CAF50;
        box-shadow: 0 0 5px rgba(76, 175, 80, 0.5);
    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }

    .error-message {
        color: #f00;
        margin-top: 5px;
        display: none;
    }
}
</style>
</head>
<body>
<div class="form-container">
    <h2>Вход</h2>
    <form id="loginForm">
        <label for="username"></label><input type="text" id="username"
placeholder="Логин">
        <br>
        <label for="password"></label><input type="password" id="password"
placeholder="Пароль">
        <br>
        <input type="submit" value="Войти">
    </form>
    <p>Нет аккаунта? <a href="registration.html">Зарегистрироваться</a></p>
    <div class="error-message" id="loginError">Введите корректные данные</div>
</div>
<script src="js/login.js"></script>
<script>
    const loginForm = document.getElementById('loginForm');
    const usernameInput = document.getElementById('username');
    const passwordInput = document.getElementById('password');
    const loginError = document.getElementById('loginError');

    loginForm.addEventListener('submit', function(event) {
        event.preventDefault();
        if (!validateLogin(usernameInput.value, passwordInput.value)) {
            loginError.style.display = 'block';
        } else {
            // Simulate successful login
            alert('Успешный вход!');
            window.location.href = 'main.html';
        }
    });

    function validateLogin(username, password) {
        // В данном случае просто проверим, что поля не пустые
        return username.trim() !== '' && password.length > 4;
    }
</script>
<script></script>
</body>
</html>

```

Файл profile.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Личный кабинет</title>
  <style>
    body {
      margin: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
      background-color: #f0f0f0;
      font-family: Arial, sans-serif;
    }
    .form-container {
      text-align: center;
      padding: 20px;
      border-radius: 10px;
      background-color: #fff;
      box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
      width: 300px;
      padding: 10px;
      margin: 10px 0;
      border-radius: 5px;
      border: 1px solid #ccc;
      font-size: 16px;
    }
    input[type="submit"] {
      background-color: #4CAF50;
      color: white;
      cursor: pointer;
    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
      /* ... (предыдущие стили) ... */
      transition: all 0.3s ease;
    }

    input[type="text"]:focus,
    input[type="password"]:focus {
      border-color: #4CAF50;
      box-shadow: 0 0 5px rgba(76, 175, 80, 0.5);
    }

    input[type="submit"]:hover {
      background-color: #45a049;
    }

    .error-message {
      color: #f00;
      margin-top: 5px;
      display: none;
    }

    .profile-info {
```



```

    const currentUser = 'user123'; // Предполагаем, что у вас есть текущий
пользователь
    const userPlaylists = JSON.parse(localStorage.getItem(currentUser)) || {};
    const playlistContainer = document.getElementById('playlistContainer');

    for (let playlist in userPlaylists) {
        const li = document.createElement('li');
        li.textContent = playlist;
        playlistContainer.appendChild(li);
    }
</script>
<script>
    function createPlaylist() {
        // Добавление нового плейлиста (для примера, пока только имитация)
        const newPlaylistName = prompt('Введите название плейлиста:');
        if (newPlaylistName) {
            const playlistContainer = document.querySelector('.playlist-
container');
            const newPlaylist = document.createElement('div');
            newPlaylist.classList.add('playlist-item');
            newPlaylist.textContent = newPlaylistName;
            newPlaylist.addEventListener('click', function() {
                // Здесь можно добавить логику для открытия плейлиста
                alert('Открыт плейлист: ' + newPlaylistName);
            });
            playlistContainer.appendChild(newPlaylist);
        }
    }

    function logout() {
        // Перенаправление на страницу входа (имитация выхода из аккаунта)
        window.location.href = 'login.html';
    }
</script>
</body>
</html>

```

Файл main.html:

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Музыкальный плеер</title>
    <style>
        body {
            margin: 0;
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
            background-color: #f0f0f0;
            font-family: Arial, sans-serif;
        }
        .form-container {
            text-align: center;
            padding: 20px;
            border-radius: 10px;
            background-color: #fff;
            box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);
        }
        input[type="text"],
        input[type="password"],
        input[type="submit"] {

```



```
        width: 300px;
        padding: 10px;
        margin: 10px 0;
        border-radius: 5px;
        border: 1px solid #ccc;
        font-size: 16px;
    }
    input[type="submit"] {
        background-color: #4CAF50;
        color: white;
        cursor: pointer;
    }
    input[type="text"],
    input[type="password"],
    input[type="submit"] {
        /* ... (предыдущие стили) ... */
        transition: all 0.3s ease;
    }

    input[type="text"]:focus,
    input[type="password"]:focus {
        border-color: #4CAF50;
        box-shadow: 0 0 5px rgba(76, 175, 80, 0.5);
    }

    input[type="submit"]:hover {
        background-color: #45a049;
    }

    .error-message {
        color: #f00;
        margin-top: 5px;
        display: none;
    }

    .music-list {
        list-style: none;
        padding: 0;
    }

    .music-item {
        display: flex;
        align-items: center;
        margin-bottom: 10px;
        cursor: pointer;
    }

    .music-item img {
        width: 50px;
        height: 50px;
        margin-right: 10px;
        border-radius: 5px;
    }

    .search-bar {
        margin-bottom: 20px;
    }

    .modal {
        display: none;
        position: fixed;
        z-index: 1;
        left: 0;
        top: 0;
        width: 100%;
        height: 100%;
        overflow: auto;
```

```

        background-color: rgba(0, 0, 0, 0.4);
    }

    .modal-content {
        background-color: #fefefe;
        margin: 15% auto;
        padding: 20px;
        border: 1px solid #888;
        width: 80%;
    }
</style>
</head>
<body>
<div class="form-container">
    <h2>Музыкальный плеер</h2>
    <a href="profile.html">Перейти в профиль</a>
    <div class="search-bar">
        <label for="searchInput"></label><input type="text" id="searchInput"
placeholder="Поиск по названию">
        <label for="genreFilter"></label><select id="genreFilter">
            <option value="all">Все жанры</option>
            <option value="rock">Рок</option>
            <option value="pop">Поп</option>
            <option value="hiphop">Хип-хоп</option>
            <!-- Другие варианты жанров -->
        </select>
        <button onclick="filterMusic()">Применить фильтр</button>
    </div>
    <ul class="music-list">
        <li class="music-item" onclick="playMusic('music/Imagine_Dragons_-_Natural_57429538.mp3')">
            
            <span>Song 1 - Rock</span>
        </li>
        <li class="music-item"
onclick="playMusic('music/DJ_Snake_Justin_Bieber_-_Let_Me_Love_You_47962926.mp3')">
            
            <span>Song 2 - Pop</span>
        </li>
        <li class="music-item"
onclick="playMusic('music/the_witcher_3_wild_hunt_16_Kaer_Morhen.mp3')">
            
            <span>Song 3 - Rock</span>
        </li>
        <!-- Добавьте другие песни -->
    </ul>
    <div id="playlistModal" class="modal">
        <div class="modal-content">
            <span class="close">&times;</span>
            <h3>Выберите плейлист</h3>
            <label for="modalPlaylistSelect"></label><select
id="modalPlaylistSelect"></select>
            <button onclick="addToSelectedPlaylist()">Добавить в
плейлист</button>
        </div>
    </div>
    <audio id="audioPlayer" controls></audio>
</div>
<script src="js/main.js"></script>
<script>
    function addToPlaylist(song) {
        const selectedPlaylist =
document.getElementById('playlistSelect').value;
        const currentUser = 'user123'; // Предполагаем, что у вас есть текущий
пользователь
    }

```

```

    let userPlaylists = JSON.parse(localStorage.getItem(currentUser)) || {};

    if (userPlaylists[selectedPlaylist]) {
        userPlaylists[selectedPlaylist].push(song);
    } else {
        userPlaylists[selectedPlaylist] = [song];
    }

    localStorage.setItem(currentUser, JSON.stringify(userPlaylists));

    alert('Песня добавлена в выбранный плейлист!');
}

// Предположим, что эта часть кода будет использоваться для загрузки
// плейлистов пользователя из профиля
// Пока оставим для наглядности
const currentUser = 'user123'; // Предполагаем, что у вас есть текущий
// пользователь
const userPlaylists = JSON.parse(localStorage.getItem(currentUser)) || {};
const playlistSelect = document.getElementById('playlistSelect');

for (let playlist in userPlaylists) {
    const option = document.createElement('option');
    option.value = playlist;
    option.textContent = playlist;
    playlistSelect.appendChild(option);
}
</script>

<script>
    function playMusic(song) {
        const audioPlayer = document.getElementById('audioPlayer');
        audioPlayer.src = song;
        audioPlayer.play();
    }

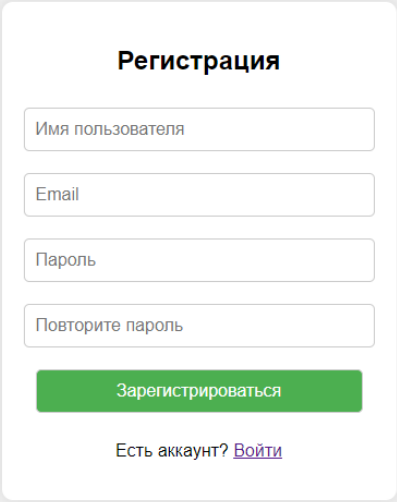
    function filterMusic() {
        const searchInput =
document.getElementById('searchInput').value.toLowerCase();
        const genreFilter =
document.getElementById('genreFilter').value.toLowerCase();
        const musicItems = document.querySelectorAll('.music-item');

        musicItems.forEach(function(item) {
            const songInfo = item.textContent.toLowerCase();
            const songGenre = songInfo.includes(genreFilter) || genreFilter ===
'all';
            const songMatch = songInfo.includes(searchInput) || searchInput ===
'';

            if (songGenre && songMatch) {
                item.style.display = 'flex';
            } else {
                item.style.display = 'none';
            }
        });
    }
</script>
</body>
</html>

```

Скриншоты страниц в браузере:
-Регистрация:



The screenshot shows a registration form titled "Регистрация" (Registration) centered on a light gray background. The form is a white card with rounded corners and a subtle shadow. It contains four input fields: "Имя пользователя" (Username), "Email", "Пароль" (Password), and "Повторите пароль" (Repeat password). Below these fields is a green button labeled "Зарегистрироваться" (Register). At the bottom, there is a link "Есть аккаунт? Войти" (Have an account? Log in).

Регистрация

Имя пользователя

Email

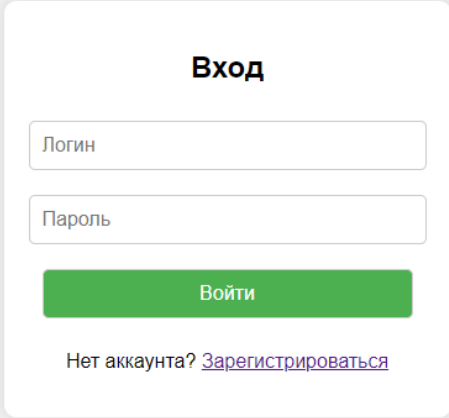
Пароль

Повторите пароль

Зарегистрироваться

Есть аккаунт? [Войти](#)

-Вход:



The screenshot shows a login form titled "Вход" (Login) centered on a light gray background. The form is a white card with rounded corners and a subtle shadow. It contains two input fields: "Логин" (Login) and "Пароль" (Password). Below these fields is a green button labeled "Войти" (Log in). At the bottom, there is a link "Нет аккаунта? Зарегистрироваться" (No account? Register).

Вход

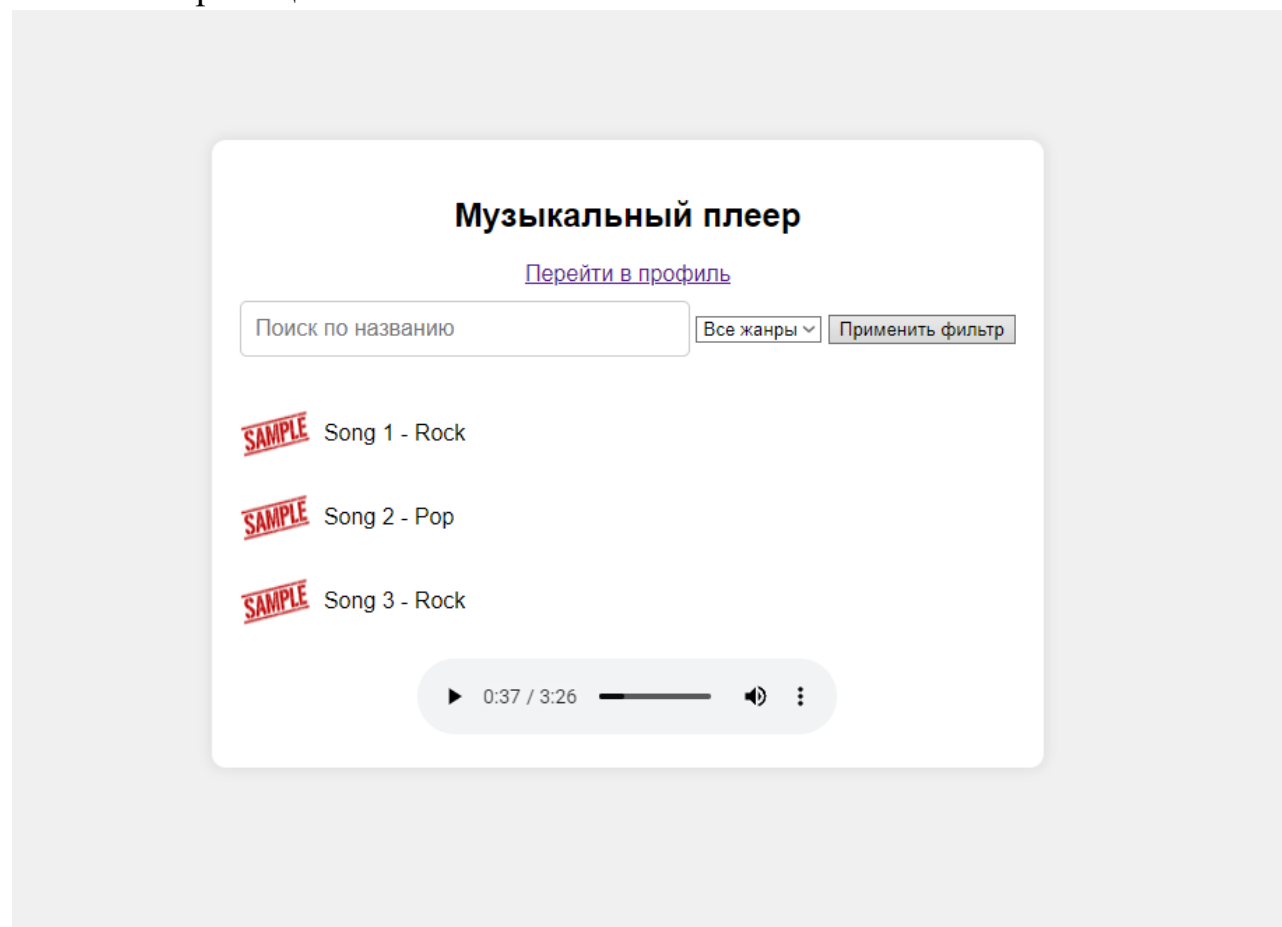
Логин

Пароль

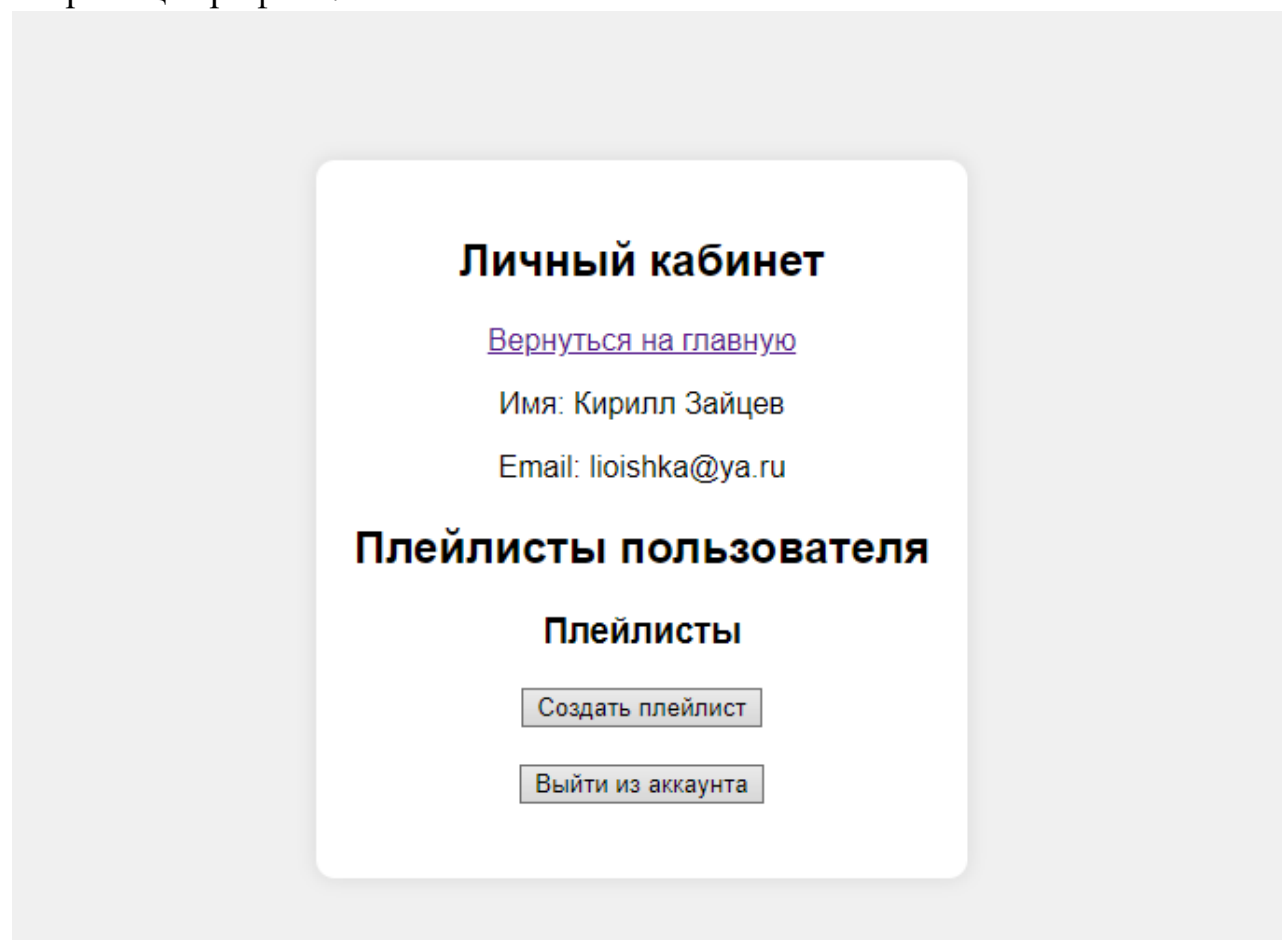
Войти

Нет аккаунта? [Зарегистрироваться](#)

-Главная страница:



-Страница профиля:



Все кнопки на каждой странице кликабельны и выполняют свой функционал(насколько возможно в рамках написания только клиентской части).

Вывод :

Работа по созданию музыкального плеера веб-приложения включала ряд ключевых этапов:

- Вёрстка сайта: Создание пользовательского интерфейса с использованием HTML, CSS и Bootstrap для обеспечения удобства и эстетики.
- Интерактивные элементы с JS: Реализация функциональности с помощью JavaScript для управления основными возможностями приложения:
- Поиск и фильтрация: Возможность поиска треков по названию и фильтрация по жанру музыки.
- Создание плейлистов: Возможность создавать новые плейлисты.
- Контроль громкости: Управление громкостью воспроизведения музыки.
- Модальные окна: Открытие модальных окон для удобного выбора плейлиста при добавлении трека.
- Реакция на действия пользователя: Создание обратной связи и уведомлений для пользователя, таких как сообщения об успешном добавлении трека или удалении плейлиста.

В результате проделанной работы получилось веб-приложение, которое обеспечивает возможность прослушивания музыки, управление плейлистами, добавление и удаление треков, а также позволяет пользователям сохранять свои настройки музыкального контента между сессиями.