

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ  
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
«КУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Факультет физики, математики, информатики

Кафедра программного обеспечения и администрирования информационных  
систем

ЛАБОРАТОРНАЯ РАБОТА №4

по дисциплине

Интерфейсы программирования приложений

на тему: *Проектирование и реализация модели поведения в веб-приложениях*

Обучающегося 2 курса очной формы  
обучения

направления подготовки

09.03.01 Информатика и вычислительная  
техника

Направленность (профиль)

Прикладной искусственный интеллект

Никитина Андрея Андреевича

Руководитель:

старший преподаватель кафедры ПОАИС

Ураева Елена Евгеньевна

Курск, 2025

## **Индивидуальное задание**

1. Используя веб-интерфейс пользователя, спроектированный в лабораторной работе 3, реализовать моделирование действий пользователя, согласно сценариям, описанным в лабораторной 1. Допустимо использовать статичные данные для элементов ввода\вывода.

2. Реализовать систему оповещений для альтернативных веток сценария.

3. В процессе реализации модели поведения веб-приложения необходимо следовать указаниям:

- по крайней мере на одной из веб-страниц интерфейса необходимо реализовать форму, содержащую не менее 5 элементов;

- для проверки корректности заполнения данных формы необходимо реализовать и использовать по крайней мере один JavaScript;

- использовать обращение к элементам формы через методы DOM (поиск, доступ), при этом реализовать по крайней мере один случай динамического наполнения страницы (изображение, текст, элементы document) за счет вызова обработчика событий;

- реализовать по крайней мере одно событие (для мыши или клавиатуры) для моделирования действия пользователя согласно сценария;

- реализовать всплывающие информационные окна и диалоговые окна для альтернативных веток сценария (можно также использовать всплывающее окно для ввода данных);

- использовать по крайней мере один JavaScript с внешним подключением.

## Интерфейс веб-клиента

На рисунках 1-8 представлена реализация модели поведения веб-приложения на JS.

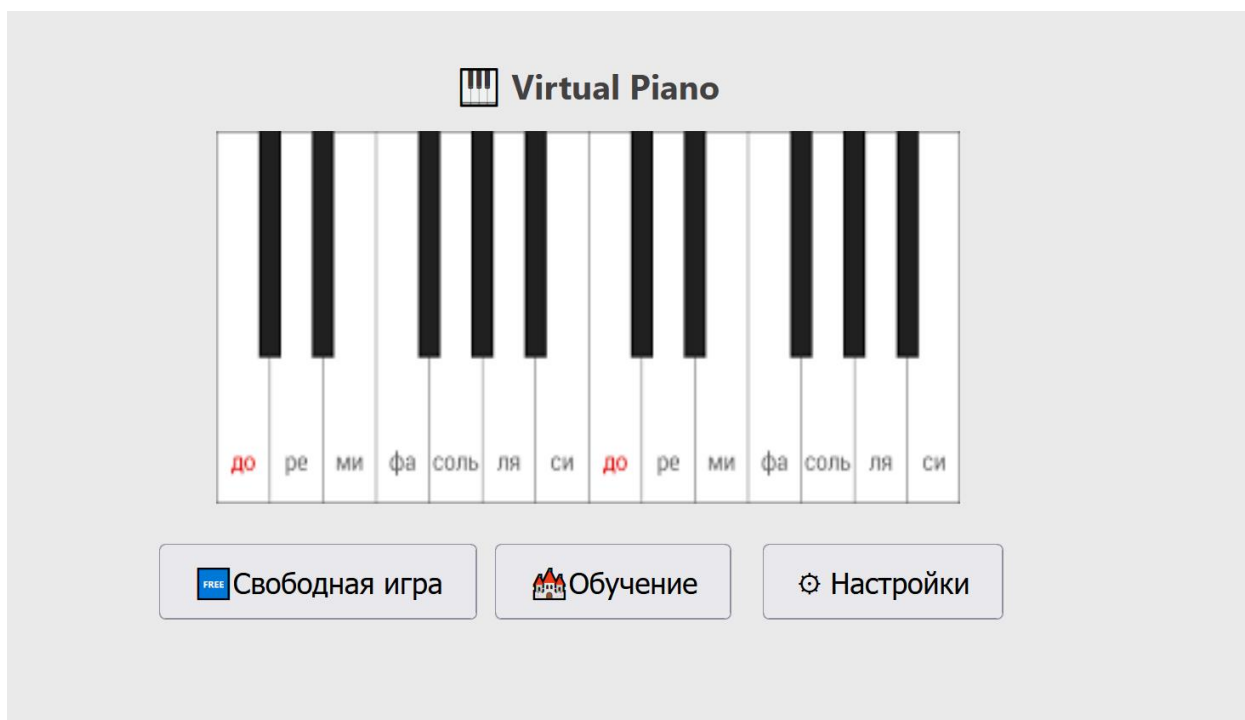


Рисунок 1 – Главная страница сайта

The image shows a feedback form titled 'Обратная связь' (Feedback) with a speech bubble icon. The form contains the following fields: 'Имя:' (Name) with a text input field, 'Email:' with a text input field, 'Любимый инструмент:' (Favorite instrument) with a dropdown menu showing 'Фортепиано' (Piano), and 'Комментарий:' (Comment) with a large text area. Below these fields is a checkbox labeled 'Я согласен с политикой конфиденциальности' (I agree with the privacy policy). At the bottom of the form is a button labeled 'Отправить' (Send).

Рисунок 2 – Форма на главной странице сайта

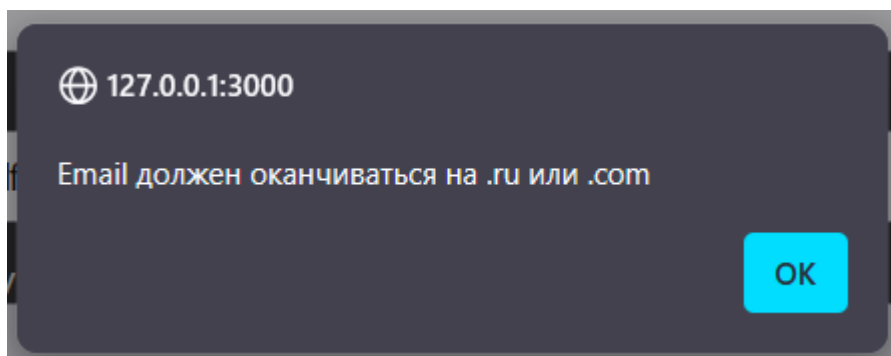


Рисунок 3 – Проверка заполнения формы

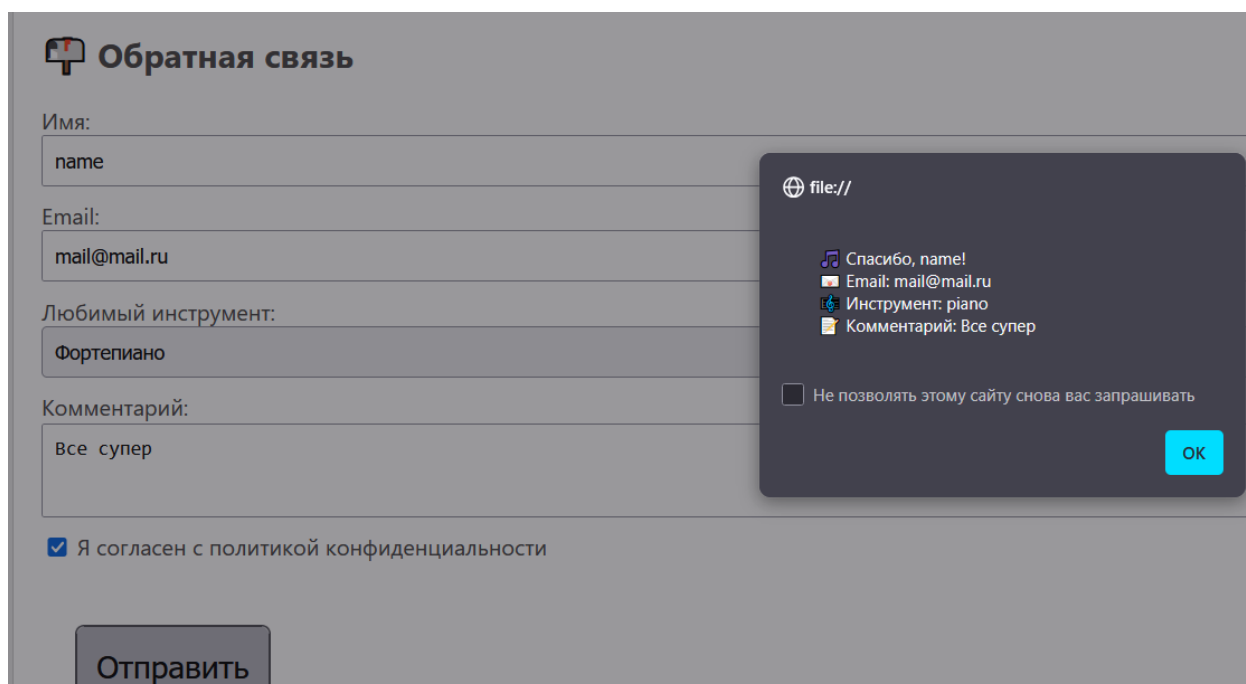


Рисунок 4 – Обработка формы

Спасибо за отзыв, name! Мы свяжемся с вами по email: mail@mail.ru.

Рисунок 5 – Динамичное наполнение страницы при вводе формы

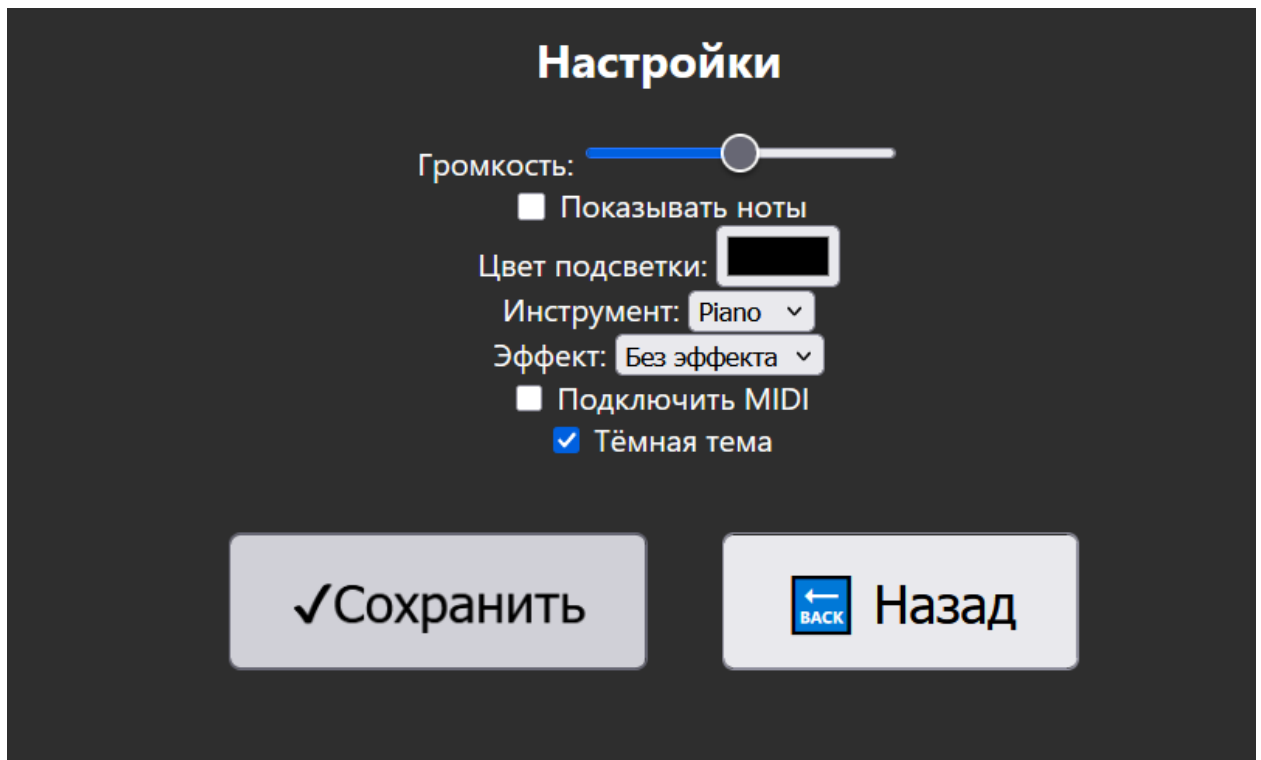


Рисунок 6 – Динамичное наполнение страницы при включении темной темы

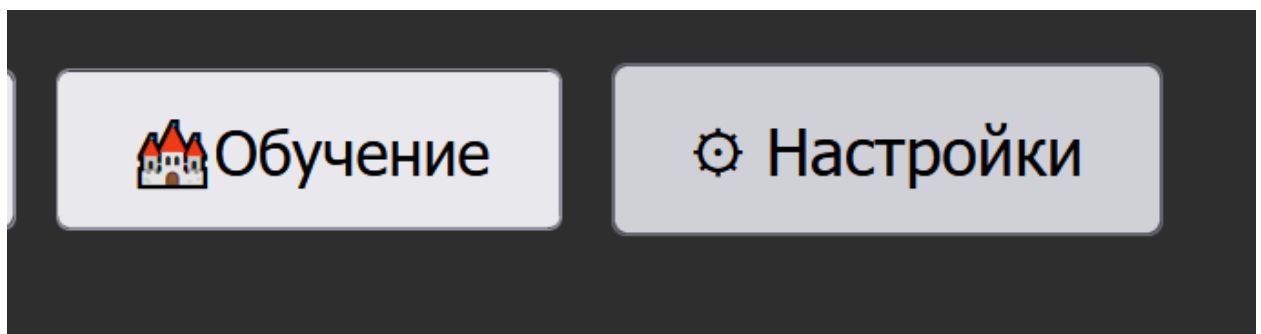


Рисунок 7 – Событие при наведении мыши на кнопку (увеличение)

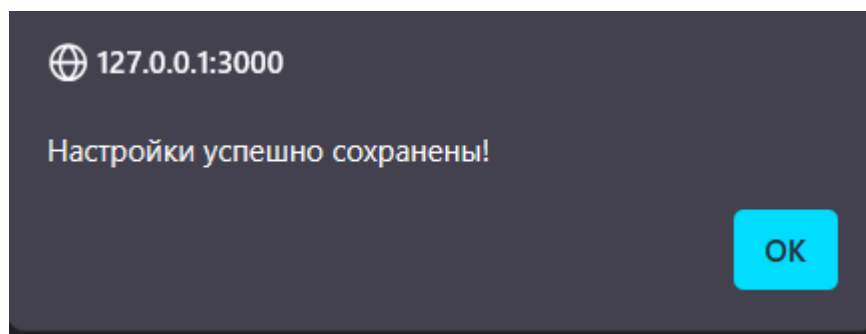





Рисунок 8 – Всплывающее окно

**Проект в среде разработки**

Ниже представлен код веб-приложения в среде разработки HTML и его модель поведения на JS:

Index.html :

```
<html lang="ru">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Virtual Piano</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <div class="main-menu">
    <h1> Virtual Piano</h1>
    
    <br>
    <button onclick="location.href='free-play.html'"> Свободная
игра</button>
    <button
onclick="location.href='learning.html'"> Обучение</button>
    <button onclick="location.href='settings.html'">
Настройки</button>
  </div>

<section id="feedbackSection" class="feedback-section">
  <h2> Обратная связь</h2>
  <form id="feedbackForm">
    <label for="name">Имя:</label>
    <input type="text" id="name" required />
```

```

<label for="email">Email:</label>
<input type="email" id="email" required />

<label for="instrument">Любимый инструмент:</label>
<select id="instrument">
  <option value="piano">Фортепиано</option>
  <option value="violin">Скрипка</option>
  <option value="guitar">Гитара</option>
  <option value="drums">Барабаны</option>
</select>

<label for="comment">Комментарий:</label>
<textarea id="comment" rows="3"></textarea>

<label>
  <input type="checkbox" id="consent" />
  Я согласен с политикой конфиденциальности
</label>
<div class="button">
  <button type="submit">Отправить</button>
</div>
</form>
</section>

<script src="script.js"></script>
<script src="form.js"></script>
</body>
</html>

```

### Free-play.html :

```

<html lang="ru">
<head>
  <meta charset="UTF-8">

```

```

<title>Свободная игра</title>
<link rel="stylesheet" href="style.css">
</head>
<body>
  <div class="controls">
    <h2>Свободная игра</h2>
    <p class="photo">
      
    </p>
    <button onclick="startRecording()"><img alt="microphone icon" data-bbox="615 310 640 330"/> Начать запись</button>
    <button onclick="stopRecording()"><img alt="stop icon" data-bbox="600 340 615 355"/> Остановить</button>
    <button onclick="playRecording()"><img alt="play icon" data-bbox="600 365 615 380"/> Воспроизвести</button>
    <button onclick="location.href='index.html'"><img alt="back icon" data-bbox="900 395 925 410"/>
Назад</button>
  </div>
  <script src="script.js"></script>
</body>
</html>

```

### learning.html :

```

<html lang="ru">
<head>
  <meta charset="UTF-8">
  <title>Обучение</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <div class="controls">
    <h2>Режим обучения</h2>
    <label for="lesson">Выберите урок:</label>
    <select id="lesson">

```



```

        <option>Гамма С-мажор</option>
        <option>Тренажер нот</option>
        <option>Детская песенка</option>
    </select>
</div>
<br>
<p class="photo">
    
</p>
<br>
<div class="controls">
    <button onclick="startLesson()">▶ Старт</button>
    <button onclick="location.href='index.html'">⬅️ Назад</button>
</div>
<script src="script.js"></script>
</body>
</html>

```

### settings.html :

```

<!DOCTYPE html>
<html lang="ru">
<head>
    <meta charset="UTF-8">
    <title>Настройки</title>
    <link rel="stylesheet" href="style.css">
</head>
<body>
    <div class="controls">
        <h2>Настройки</h2>
        <label>Громкость: <input type="range" id="volume" min="0"
max="1" step="0.01"></label><br>

```

```

        <label><input type="checkbox" id="showNotes"> Показывать
ноты</label><br>
        <label>Цвет подсветки: <input type="color"
id="highlightColor"></label><br>
        <label>Инструмент:
        <select id="instrument">
            <option value="piano">Piano</option>
            <option value="organ">Organ</option>
            <option value="synth">Synth</option>
        </select>
    </label><br>
    <label>Эффект:
        <select id="effect">
            <option value="none">Без эффекта</option>
            <option value="reverb">Reverb</option>
            <option value="echo">Echo</option>
        </select>
    </label><br>
        <label><input type="checkbox" id="midi"> Подключить
MIDI</label><br>
        <label><input type="checkbox" id="theme"> Тёмная
тема</label><br>
        <button onclick="saveSettings()">✓Сохранить</button>
        <button onclick="location.href='index.html'">🏠 Назад</button>
    </div>
    <script src="script.js"></script>
</body>
</html>

```

style.css :

```

body {
    font-family: 'Segoe UI', sans-serif;

```

```
background-color: #eaeaea;
margin: 0;
padding: 30px;
color: #434242;
}
```

```
button {
  margin: 50px;
  padding: 10px 20px;
  font-size: 18px;
  cursor: pointer;
  transform: scale(1.5);
}
```

```
.main-menu {
  text-align: center;
}
```

```
.piano {
  display: flex;
  justify-content: center;
  margin: 20px auto;
  max-width: 900px;
}
```

```
.controls {
  text-align: center;
}
```

```
.photo{
  text-align: center;
}
```

```
.set{
  text-align: justify;
```

```

}
.feedback-section {
  background-color: #fff;
  padding: 20px;
  margin: 30px;
  border: 1px solid #ccc;
  border-radius: 8px;
}

.feedback-section form {
  display: flex;
  flex-direction: column;
}

.feedback-section input,
.feedback-section select,
.feedback-section textarea,
.feedback-section button {
  margin-bottom: 10px;
  padding: 8px;
  font-size: 14px;
}

```

### Script.js :

```

function saveSettings() {
  const instrument = document.getElementById("instrument").value;
  const volume =
parseFloat(document.getElementById("volume").value);
  const showNotes = document.getElementById("showNotes").checked;
  const highlightColor =
document.getElementById("highlightColor").value;
  const theme = document.getElementById("theme").checked;
  const effect = document.getElementById("effect").value;
  const midi = document.getElementById("midi").checked;
}

```

```

if (!instrument) {
    alert("Пожалуйста, выберите инструмент!");
    return;
}

const settings = {
    instrument,
    volume,
    showNotes,
    highlightColor,
    theme,
    effect,
    midi
};

                                localStorage.setItem("pianoSettings",
JSON.stringify(settings));

    alert("Настройки успешно сохранены!");

    applyTheme(theme);
}

function applySettings() {
    const saved = localStorage.getItem("pianoSettings");
    if (!saved) return;

    const settings = JSON.parse(saved);

    applyTheme(settings.theme);

    console.log("Цвет подсветки:", settings.highlightColor);

    console.log("Громкость:", settings.volume);

```

```

    console.log("Эффект:", settings.effect);

    console.log("Инструмент:", settings.instrument);
}

function applyTheme(isDark) {
    if (isDark) {
        document.body.style.backgroundColor = "#2e2e2e";
        document.body.style.color = "white";

        document.getElementById("feedbackSection").style.backgroundC
olor = "#262626";
        document.getElementById("feedbackSection").style.color =
"white";
    } else {
        document.body.style.backgroundColor = "#eaeaea";
        document.body.style.color = "#434242";
    }
}

function addHoverEffects() {
    const buttons = document.querySelectorAll("button");
    buttons.forEach(button => {
        button.addEventListener("mouseover", () => {
            button.style.transform = "scale(1.6)";
            button.style.transition = "transform 0.2s";
        });
        button.addEventListener("mouseout", () => {
            button.style.transform = "scale(1.5)";
        });
    });
}

function startLesson() {

```

```

const lesson = document.getElementById("lesson").value;
if (confirm(`Вы действительно хотите начать урок "${lesson}"?`))
{
    alert(`Урок "${lesson}" начался!`);
} else {
    alert("Отмена.");
}
}

document.addEventListener("DOMContentLoaded", () => {
    applySettings();
    addHoverEffects();
});

```

### Form.js :

```

document.addEventListener('DOMContentLoaded', () => {
    const form = document.getElementById('feedbackForm');

    form.addEventListener('submit', function (e) {
        e.preventDefault();

        const name = document.getElementById('name').value.trim();
        const email = document.getElementById('email').value.trim();
        const instrument = document.getElementById('instrument').value;
        const comment = document.getElementById('comment').value.trim();
        const consent = document.getElementById('consent').checked;

        if (!name || !email || !consent) {
            alert('Пожалуйста, заполните имя, email и примите политику.');
```

```

const emailPattern = /^[^@]+@[^@]+\.(ru|com)$/i;
if (!emailPattern.test(email)) {
    alert('Email должен оканчиваться на .ru или .com');
    return;
}

    if (!confirm(`Вы уверены, что хотите отправить сообщение от
${name}?`)) {
        return;
    }

const message = `
    🎵 Спасибо, ${name}!
    ✉ Email: ${email}
    🎸 Инструмент: ${instrument}
    📝 Комментарий: ${comment || "без комментариев"}
`;

alert(message);

const feedbackSection =
document.getElementById('feedbackSection');
const response = document.createElement('p');
    response.textContent = `Спасибо за отзыв, ${name}! Мы
свяжемся с вами по email: ${email}.`;
feedbackSection.appendChild(response);

form.reset();
});
});

```