

**Разработка интерактивного веб-приложения
для обучения школьников арифметическим операциям
с использованием JavaScript и HTML5 Canvas**

Кокухина Софья Александровна

Приамурский государственный университет им.Шолом-Алейхема

Студент

Аннотация

В данной статье рассматривается процесс разработки интерактивного веб-приложения для обучения школьников арифметическим операциям с использованием HTML5 Canvas и JavaScript. Описываются этапы создания приложения, включая проектирование интерфейса, реализацию логики генерации примеров и визуальной обратной связи с помощью анимации. Особое внимание уделяется простоте реализации и адаптации приложения для образовательных целей, а также практическим рекомендациям по созданию интерактивных инструментов, способствующих повышению интереса школьников к математике.

Ключевые слова: Обучение математике, веб-приложение, HTML5 Canvas, JavaScript, интерактивное обучение, школьное образование, арифметические операции.

**Development of an interactive web application for teaching students
arithmetic operations using JavaScript and HTML5 Canvas**

Kokukhina Sofya Alexandrovna

Sholom-Aleichem Priamursky State University

Student

Abstract

This article discusses the process of developing an interactive web application for teaching students' arithmetic operations using HTML5 Canvas and JavaScript. The stages of application creation are described, including interface design, implementation of logic for generating examples and visual feedback using animation. Special attention is paid to the ease of implementation and adaptation of the application for educational purposes, as well as practical recommendations for creating interactive tools that help to increase students' interest in mathematics.

Keywords Teaching mathematics, web application, HTML5 Canvas, JavaScript, interactive learning, school education, arithmetic operations.

1. Введение

1.1 Актуальность

С развитием информационных технологий и цифровизацией образования возрастает потребность в интерактивных инструментах, которые делают изучение математики увлекательным и доступным для школьников. Традиционные методы обучения часто не обеспечивают достаточной мотивации, особенно для учеников начальной и средней школы. Использование веб-технологий, таких как HTML5 Canvas и JavaScript, позволяет создавать простые, но эффективные образовательные приложения, которые сочетают интерактивность и визуальную обратную связь. Однако существует недостаток структурированных и доступных руководств по созданию таких приложений, адаптированных для школьной аудитории. Данная статья восполняет этот пробел, предлагая пошаговое руководство по разработке веб-приложения для тренировки арифметических навыков, демонстрируя возможности HTML5 Canvas и JavaScript для создания образовательных инструментов.

1.2 Обзор исследований

В работах А.А. Черепанова показано, что интерактивные веб-приложения на JavaScript с использованием HTML5 Canvas позволяют эффективно визуализировать математические объекты и поддерживать исследовательскую деятельность учащихся; автор описывает web-приложение PhaPl и подчёркивает значимость динамических анимаций/рисования на Canvas для наглядности и вовлечения [1]. Г.Ю. Протоद्याконова и Л.А.Лебедева описывают разработку мультимедийного учебного комплекса для начальной и средней школы, где в качестве базовых технологий применяются HTML5 Canvas и WebGL; отмечается рост мотивации при работе с интерактивной графикой и анимацией [2]. К.Д. Горбунова и И.В. Баженова демонстрируют, как средствами HTML, CSS и JavaScript выстраивать веб-квест по математике для старших школьников: через задания, обратную связь и сценарии навигации — фактически тот же каркас, что и у тренажёров арифметики в браузере [3]. Эти результаты в совокупности поддерживают идею разработки интерактивного учебного веб-приложения по арифметике на JS/Canvas: Canvas отвечает за быструю и «живую» визуализацию операций (анимация сложения/вычитания, дробление/группировка при умножении/делении), а JavaScript — за логику заданий, адаптивность и мгновенную проверку.

1.3 Цель исследования

Целью данного исследования является разработка и демонстрация интерактивного веб-приложения для обучения школьников арифметическим операциям (сложение и вычитание) с использованием HTML5 Canvas и JavaScript, направленного на повышение мотивации и вовлечённости учащихся. В рамках исследования создаётся простое приложение, которое генерирует случайные арифметические примеры, проверяет ответы

пользователей и предоставляет визуальную обратную связь в виде анимации. Рассматриваются ключевые аспекты реализации: проектирование пользовательского интерфейса, генерация заданий, обработка ответов и создание анимации на Canvas. Тестирование приложения демонстрирует, как интерактивные элементы и визуализация способствуют упрощению изучения математики. Также подчёркиваются преимущества использования HTML5 и JavaScript: доступность в браузере, простота разработки и возможность адаптации для образовательных целей.

2 Материалы и методы

Для разработки веб-приложения используется HTML5 для создания структуры страницы, CSS3 для стилизации интерфейса и JavaScript для реализации логики и интерактивности. В качестве инструмента визуализации применяется HTML5 Canvas, обеспечивающий создание анимации для обратной связи. Разработка ведётся в редакторе кода Visual Studio Code, который предоставляет удобный интерфейс и инструменты для отладки. Все компоненты приложения реализуются без использования внешних библиотек, что упрощает внедрение и обеспечивает кроссбраузерную совместимость. Тестирование проводится в современных веб-браузерах (Google Chrome, Mozilla Firefox), чтобы гарантировать корректное отображение и функциональность приложения.

2.1 Подготовка рабочей среды

Для разработки используется Visual Studio Code (VS Code) — бесплатный редактор кода с поддержкой HTML, CSS и JavaScript. После установки нужно открыть VS Code, и создать новую папку проекта с названием MathQuiz через меню File -> Open Folder.

Для удобного тестирования веб-приложения в браузере устанавливается расширение Live Server в VS Code. В разделе Extensions (Ctrl+Shift+X) вводится запрос "Live Server", выбирается расширение и нажимается кнопка Install. Это расширение позволяет запускать приложение в браузере с автоматическим обновлением при изменении кода.

В папке MathQuiz создаются три файла:

index.html — для структуры страницы.

style.css — для стилизации интерфейса.

script.js — для логики приложения. Для создания файлов в VS Code используется меню File -> New File, после чего файлы сохраняются с соответствующими именами (Рис. 1).

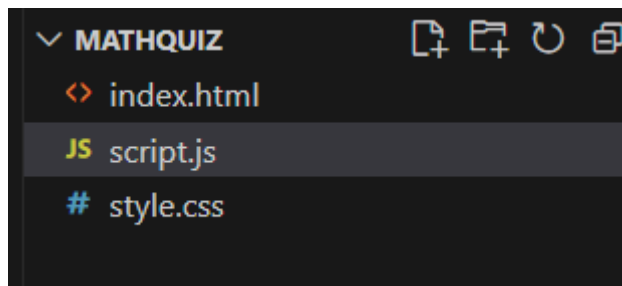


Рисунок 1 – Подготовка необходимых файлов

3 Разработка веб-приложения

В файл index.html добавляется следующий код, определяющий структуру страницы (Рис.2).

```
1  <!DOCTYPE html>
2  <html lang="ru">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Математическая викторина</title>
7      <link rel="stylesheet" href="style.css">
8  </head>
9  <body>
10     <div class="container">
11         <h1>Математическая викторина</h1>
12         <canvas id="mathCanvas" width="400" height="200"></canvas>
13         <p id="question"></p>
14         <input type="text" id="answer" placeholder="Введите ответ">
15         <button onclick="checkAnswer()">Ответить</button>
16         <p id="feedback"></p>
17     </div>
18     <script src="script.js"></script>
19 </body>
20 </html>
```

Рисунок 2 – Код файла index.html

«<!DOCTYPE html>» указывает, что документ соответствует стандарту HTML5. «<meta charset="UTF-8">» задаёт кодировку UTF-8 для корректного отображения русских символов. «<meta name="viewport">» обеспечивает адаптивность интерфейса для мобильных устройств.

В файл style.css добавляется код для оформления страницы (Рис.2).

```
1  √ body {
2      font-family: Arial, sans-serif;
3      display: flex;
4      justify-content: center;
5      background-color: #f0f0f0;
6      margin: 0;
7  }
8  √ .container {
9      text-align: center;
10     margin-top: 20px;
11     max-width: 600px;
12 }
13 √ canvas {
14     border: 2px solid #333;
15     background-color: #fff;
16     margin-bottom: 20px;
17 }
18 √ input, button {
19     margin: 10px;
20     padding: 8px;
21     font-size: 16px;
22     border-radius: 5px;
23 }
24 √ button {
25     background-color: #4CAF50;
26     color: white;
27     border: none;
28     cursor: pointer;
29 }
30 √ button:hover {
31     background-color: #45a049;
32 }
33 √ p {
34     font-size: 18px;
35 }
```

Рисунок 2 – Код стилей из style.css

В файл script.js добавляется код для генерации примеров, проверки ответов и создания анимации (Рис.3).

```

1  // Получение элементов DOM
2  const canvas = document.getElementById('mathCanvas');
3  const ctx = canvas.getContext('2d');
4  const questionElement = document.getElementById('question');
5  const answerInput = document.getElementById('answer');
6  const feedbackElement = document.getElementById('feedback');
7
8  // Функция генерации случайного примера
9  function generateQuestion() {
10     const num1 = Math.floor(Math.random() * 100) + 1; // Число от 1 до 100
11     const num2 = Math.floor(Math.random() * 100) + 1;
12     const operations = ['+', '-'];
13     const operation = operations[Math.floor(Math.random() * 2)];
14     let correctAnswer;
15     if (operation === '+') {
16         correctAnswer = num1 + num2;
17     } else {
18         correctAnswer = num1 - num2;
19     }
20     questionElement.textContent = `${num1} ${operation} ${num2} = ?`;
21     return correctAnswer;
22 }
23
24 // Инициализация первого вопроса
25 let correctAnswer = generateQuestion();
26
27 // Функция рисования звезды
28 function drawStar(x, y) {
29     ctx.beginPath();
30     ctx.moveTo(x, y - 10);
31     for (let i = 0; i < 5; i++) {
32         ctx.lineTo(x + 10 * Math.cos(Math.PI * 0.4 * i), y + 10 * Math.sin(Math.PI * 0.4 * i));
33         ctx.lineTo(x + 5 * Math.cos(Math.PI * 0.4 * (i + 0.5)), y + 5 * Math.sin(Math.PI * 0.4 * (i + 0.5)));
34     }
35     ctx.closePath();
36     ctx.fillStyle = 'yellow';
37     ctx.fill();
38 }

```

Рисунок 3 – Код скриптов проекта

После написания всех частей кода, проект готов к запуску. Для этого в VS Code щелкается правая кнопка мыши на файле index.html, выбирается Open with Live Server, и страница автоматически открывается в браузере. При загрузке страницы отображается первый арифметический пример, например, $45 + 32 = ?$. В поле ввода вводится ответ, после чего нажимается кнопка "Ответить". Если ответ верный, на Canvas отображаются жёлтые звёзды, а текст "Правильно!" появляется в элементе `<p id="feedback">`. Если ответ неверный, отображается сообщение с правильным ответом, например, "Неправильно! Правильный ответ: 77". После каждого ответа генерируется новый пример (Рис. 4).

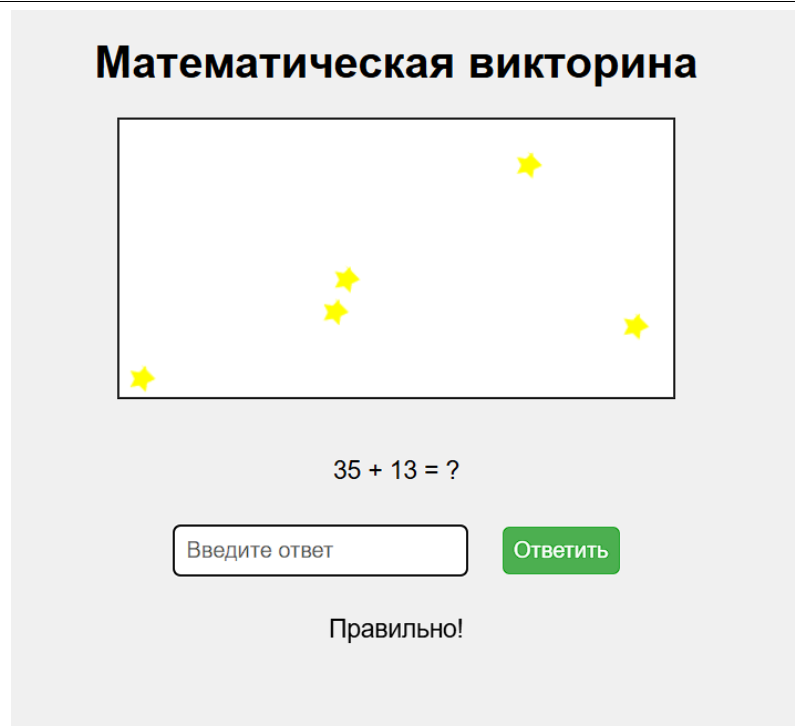


Рисунок 4 – Окно викторины в браузере

Также можно убедиться, что приложение корректно работает даже в случае неверного числа. Ввод некорректных данных (например, текста или пустого поля), что вызывает сообщение "Введите число!" (Рис.5).

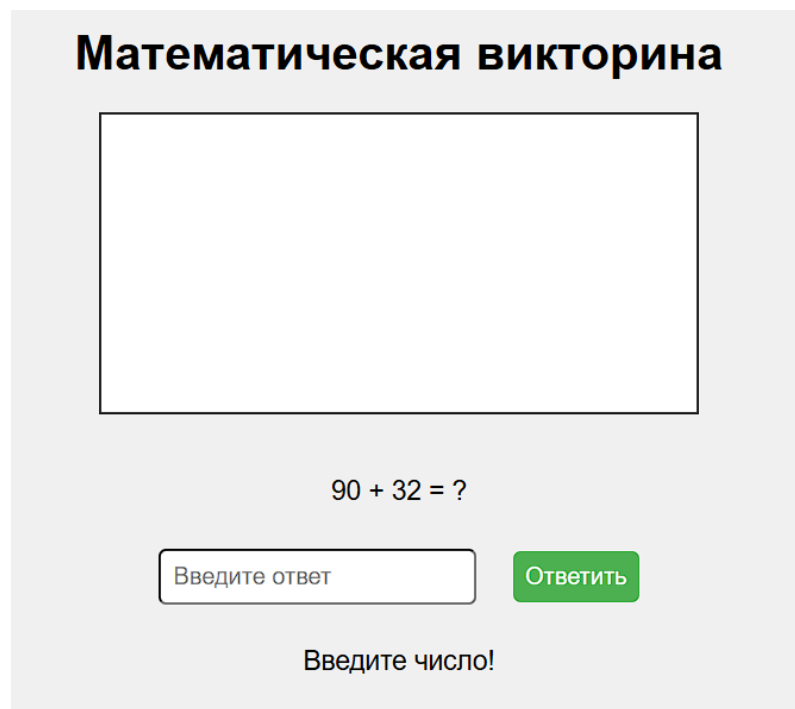


Рисунок 5 – Сообщение ошибки

4 Выводы

В ходе практической части разработано веб-приложение для обучения школьников арифметическим операциям. Использование HTML5 Canvas и

JavaScript позволило создать простой, но эффективный инструмент с интерактивным интерфейсом и визуальной обратной связью. Приложение успешно протестировано, демонстрируя корректную работу генерации примеров, проверки ответов и анимации. Простота кода делает его доступным для реализации учителями и начинающими разработчиками, а возможность доработки (например, добавление новых операций или сохранение результатов) обеспечивает гибкость для образовательных целей. Результаты подтверждают, что интерактивные веб-приложения могут значительно повысить интерес школьников к математике

Библиографический список

1. Черепанов А. А. Веб-приложение PhaPI для автоматического построения и исследования фазовых портретов на плоскости // Открытое образование. 2017. № 3. С. 66–72. URL: <https://journals.ssau.ru/est/article/view/6452> (дата обращения: 13.08.2025).
2. Протоद्याконова Г. Ю., Лебедева Л. А. Разработка мультимедийного комплекса для обучения учащихся в начальной и средней школе // К.Д.Ушинский и национальное образование: материалы науч.-практ. конф. М., 2021. URL: <https://elibrary.ru/item.asp?id=46447547> (дата обращения: 13.08.2025).
3. Горбунова К. Д., Баженова И. В. Реализация web-квеста по математике для учащихся старшей школы средствами HTML, CSS и JavaScript // Актуальные проблемы авиации и космонавтики: материалы Междунар. науч.-практ. конф. Красноярск: СибГАУ, 2015. С. 1171–1173. URL: <https://cyberleninka.ru/article/n/realizatsiya-web-kvesta-po-matematike-dlya-uchaschihsya-starshey-shkoly-sredstvami-html-css-i-javascript> (дата обращения: 13.08.2025).