

Создание образовательного веб-теста по информатике

Анишкова Анастасия Сергеевна

Приамурский государственный университет имени Шолом-Алейхема

Студент

Аннотация

Целью работы является разработать интерактивный, удобный и эффективный с точки зрения обучения веб-тест по информатике с использованием технологий HTML, CSS и JavaScript, который позволит проверять знания учащихся, обеспечивать мгновенную обратную связь, контролировать время выполнения заданий и сохранять историю попыток для последующего анализа успеваемости. Для реализации выбран чистый JavaScript (Vanilla JS) в сочетании с HTML5 и CSS3 без использования сторонних библиотек и фреймворков. Такой подход обеспечивает высокую производительность и минимальную нагрузку на систему, позволяя приложению быстро загружаться и стабильно работать на различных устройствах, включая мобильные. Отказ от внешних зависимостей даёт полный контроль над логикой работы теста, управлением состоянием интерфейса и обработкой пользовательских данных. Реализация на HTML, CSS и JavaScript без использования внешних библиотек делает его легковесным и доступным для использования в образовательной среде.

Ключевые слова: JavaScript, тест по информатике, веб-тест, HTML.

Creating an Educational Web-Based Computer Science Test

Anishkova Anastasia Sergeevna

Sholom-Aleichem Priamurskiy State University

Student

Abstract

The goal of this work is to develop an interactive, user-friendly, and effective web-based computer science test using HTML, CSS, and JavaScript technologies. This test will allow students to test their knowledge, provide instant feedback, control the time it takes to complete tasks, and save their attempt history for later analysis of their performance. To achieve this goal, we have chosen to use pure JavaScript (Vanilla JS) in combination with HTML5 and CSS3, without the use of third-party libraries or frameworks. This approach ensures high performance and minimal system load, allowing the application to load quickly and run smoothly on various devices, including mobile devices. Abandoning external dependencies gives full control over the test logic, interface state management, and user data processing. Implemented in HTML, CSS, and JavaScript without external libraries, it is lightweight and accessible for use in an educational environment.

Keywords: JavaScript, computer science test, web test, HTML.

1 Введение

1.1 Актуальность

Актуальность данного теста по информатике обусловлена возрастающей ролью цифровой грамотности в современном образовании. Интерактивный формат с таймером и мгновенной обратной связью соответствует требованиям к современным образовательным платформам, обеспечивая объективную оценку базовых IT-знаний. Система сохранения результатов и их анализа удовлетворяет потребность в отслеживании прогресса обучения, а адаптивный интерфейс отвечает трендам на мобильное и доступное образование.

1.2 Обзор исследований

В статье рассматривается организация подготовки педагогических работников к использованию технологий HTML5 для разработки электронных образовательных ресурсов в формате веб-приложений. Отмечена актуальность использования мультимедийных интерактивных электронных образовательных ресурсов в современной школе и обоснована эффективность разработки подобных электронных образовательных ресурсов именно в формате офлайновых или онлайновых веб-приложений [1]. Показала процесс создания и использования мультимедийных интерактивных плакатов по информатике К.А. Цилинская. Представила опыт разработки мультимедийных интерактивных плакатов средствами языка HTML5 и Интернет-сервисов [2]. Королюк Я. А., Куликова Н. Ю. показали опыт разработки и использования интерактивных образовательных ресурсов на основе html5 для онлайн-обучения школьников информатике [3]. В работе рассмотрели возможности языка разметки гипертекста HTML5 для создания макета электронного учебника Д.М. Размологов, А.С. Панина, П.В. Захаров. Привели разработанный макет, а также часть таблицы стилей для дизайна макета [4]. И.С. Шегаев, А.Г. Полозов разработали новейший политологический глоссарий: основные категории, понятия, термины [5].

1.3 Цель исследования

Целью работы является разработать интерактивный, удобный и эффективный с точки зрения обучения веб-тест по информатике с использованием технологий HTML, CSS и JavaScript, который позволит проверять знания учащихся, обеспечивать мгновенную обратную связь, контролировать время выполнения заданий и сохранять историю попыток для последующего анализа успеваемости. Для реализации выбран чистый JavaScript (Vanilla JS) в сочетании с HTML5 и CSS3 без использования сторонних библиотек и фреймворков.

2 Материалы и методы

Для реализации интерактивного теста применялись стандартные веб-технологии (HTML, CSS, JavaScript) с динамическим рендерингом контента, клиентским хранением данных и таймером обратного отсчета.

Тест можно скачать по ссылке <https://disk.yandex.ru/d/5VURdPgCU1xvjg>.

3 Результаты

Базовый каркас HTML-документа задаёт необходимые настройки для кроссплатформенного образовательного веб-приложения. В секции <head> объявлена кодировка UTF-8 для корректного отображения текста на русском языке, а атрибут viewport обеспечивает адаптивность интерфейса на мобильных устройствах. Заголовок страницы указывает на её образовательную направленность — тестирование по информатике:

```
<!DOCTYPE html>
<html lang=«ru»>
<head>
  <meta charset=«UTF-8» />
  <meta name=«viewport» content=«width=device-width, initial-
scale=1.0»/>
  <title>Тест по информатике</title>
</head>
</html>
```

Стилевое оформление тестовой платформы реализует современный, минималистичный дизайн с акцентом на удобство восприятия образовательного контента. Цветовая палитра построена на контрасте светлого фона (#f0f2f5) и чистого белого контейнера для контента, что снижает зрительную нагрузку во время prolonged тестирования. Таймер выделен предупреждающим красным цветом (#e74c3c) для визуального акцентирования временных ограничений.

Интерактивные элементы используют интуитивно понятные состояния: кнопки с плавными hover-эффектами, визуальная обратная связь для правильных/неправильных ответов через цветовую индикацию (.correct, .incorrect), и четкое разделение вопросов и вариантов ответов через продуманные отступы. История ответов оформлена в виде минималистичного журнала с цветовым кодированием и выделением через градиентные границы, что обеспечивает быстрое сканирование результатов.

Адаптивная верстка контейнера (max-width: 600px) гарантирует комфортное использование как на десктопных, так и на мобильных устройствах, а тени и скругления углов создают современный «карточный» интерфейс, соответствующий современным тенденциям UX/UI-дизайна.

```
<style>
body {
```

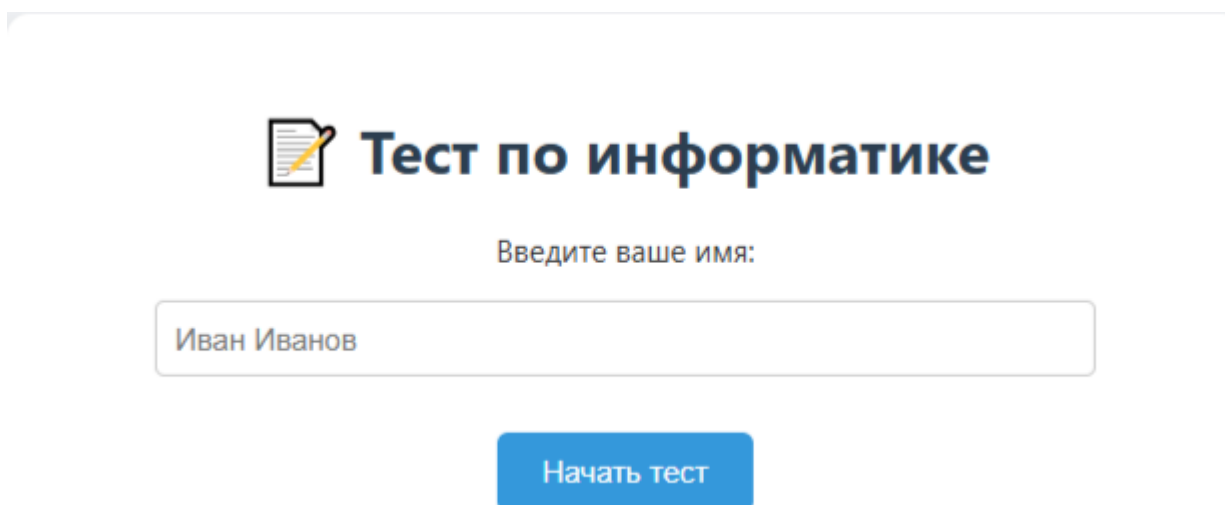
```
font-family: 'Segoe UI', sans-serif;
background-color: #f0f2f5;
color: #333;
text-align: center;
padding: 40px;
}
.container {
  max-width: 600px;
  margin: 0 auto;
  background: white;
  padding: 30px;
  border-radius: 12px;
  box-shadow: 0 4px 15px rgba(0,0,0,0.1);
}
h1 {
  color: #2c3e50;
}
.timer {
  font-size: 18px;
  font-weight: bold;
  color: #e74c3c;
  margin: 10px 0;
}
.section {
  margin: 20px 0;
}
.question {
  text-align: left;
  margin: 15px 0;
}
.options {
  margin-top: 10px;
}
.option {
  margin: 8px 0;
}
input[type=«text»] {
  padding: 10px;
  font-size: 16px;
  width: 80%;
  border: 1px solid #ccc;
  border-radius: 5px;
}
button {
  padding: 12px 24px;
```

```
font-size: 16px;
background-color: #3498db;
color: white;
border: none;
border-radius: 6px;
cursor: pointer;
margin: 10px 5px;
}
button:hover {
  background-color: #2980b9;
}
button:disabled {
  background-color: #95a5a6;
  cursor: not-allowed;
}
#result {
  margin-top: 20px;
  font-size: 18px;
  font-weight: bold;
  min-height: 60px;
}
#history {
  margin-top: 30px;
  text-align: left;
  font-size: 14px;
  color: #7f8c8d;
  border-top: 1px dashed #bdc3c7;
  padding-top: 15px;
}
#history h3 {
  margin-bottom: 10px;
  color: #2c3e50;
}
#history ul {
  list-style-type: none;
  padding: 0;
}
#history li {
  margin: 5px 0;
  padding: 5px;
  background-color: #f8f9fa;
  border-left: 3px solid #3498db;
}
.correct {
  color: #27ae60;
```

```
}  
.incorrect {  
  color: #e74c3c;  
}  
.hidden {  
  display: none;  
}  
</style>
```

Стартовый экран предлагает минималистичный интерфейс для начала тестирования (рис.1). Поле ввода имени с плейсхолдером обеспечивает удобство ввода данных. Кнопка «Начать тест» имеет увеличенную область клика согласно закону Фиттса. Эмодзи в заголовке создает тематическую ассоциацию. Четкая композиция направляет внимание пользователя от названия теста к вводу данных и действию.

```
</head>  
<body>  
  <div class=«container»>  
    <h1>📝 Тест по информатике</h1>  
  
    <!-- Ввод имени -->  
    <div id=«startScreen»>  
      <p>Введите ваше имя:</p>  
      <input type=«text» id=«studentName» placeholder=«Иван Иванов» />  
      <div class=«section»>  
        <button onclick=«startQuiz()»>Начать тест</button>  
      </div>  
    </div>  
  </div>
```



📝 Тест по информатике

Введите ваше имя:

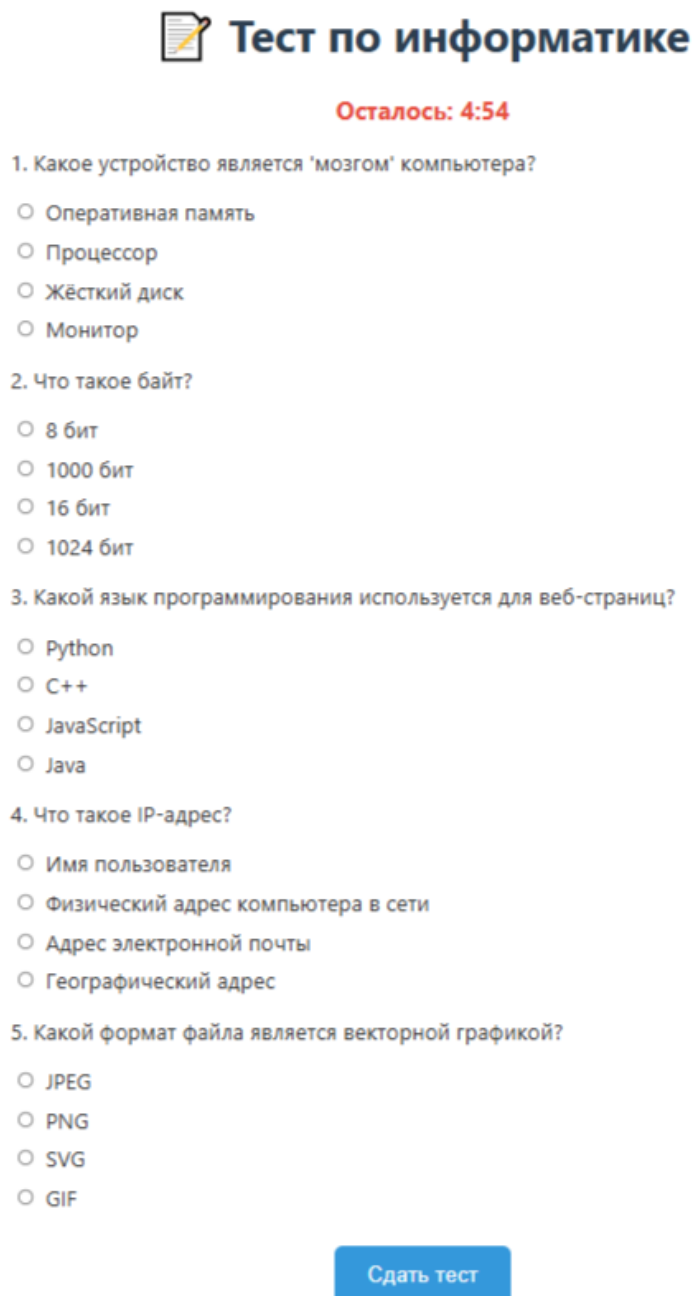
Иван Иванов

Начать тест

Рисунок 1. Стартовый экран

JavaScript-код реализует полную логику работы интерактивного тестирования. Он управляет всем жизненным циклом теста: от стартового экрана с валидацией имени пользователя до динамического отображения вопросов, обработки ответов и финального вывода результатов. Ядро системы — массив вопросов с метаданными, включая варианты ответов и индекс правильного.

Таймер обратного отсчета автоматически завершает тест по истечении времени, обеспечивая контроль длительности сессии (рис.2).



 **Тест по информатике**

Осталось: 4:54

1. Какое устройство является 'мозгом' компьютера?

- ☐ Оперативная память
- ☐ Процессор
- ☐ Жёсткий диск
- ☐ Монитор

2. Что такое байт?

- ☐ 8 бит
- ☐ 1000 бит
- ☐ 16 бит
- ☐ 1024 бит

3. Какой язык программирования используется для веб-страниц?

- ☐ Python
- ☐ C++
- ☐ JavaScript
- ☐ Java

4. Что такое IP-адрес?

- ☐ Имя пользователя
- ☐ Физический адрес компьютера в сети
- ☐ Адрес электронной почты
- ☐ Географический адрес

5. Какой формат файла является векторной графикой?

- ☐ JPEG
- ☐ PNG
- ☐ SVG
- ☐ GIF

[Сдать тест](#)

Рисунок 2. Страница теста

Реализована система оценки с персонализированными сообщениями в зависимости от процента правильных ответов. Все результаты сохраняются в

локальное хранилище браузера, что позволяет вести историю попыток с возможностью просмотра последних пяти результатов (рис.3), их экспорта в CSV-файл или полной очистки (рис.4).

○ 1000 бит
○ 16 бит
○ 1024 бит

3. Какой язык программирования используется для веб-страниц?

○ Python
○ C++
● JavaScript
○ Java

4. Что такое IP-адрес?

○ Имя пользователя
● Физический адрес компьютера в сети
○ Адрес электронной почты
○ Географический адрес

5. Какой формат файла является векторной графикой?

○ JPEG
○ PNG
● SVG
○ GIF

[Пройти снова](#)

Иванов Иван, отлично! 5 из 5 правильных ответов!

Ваши предыдущие результаты:

Иванов Иван: 5/5 (100%) — 27.08.2025, 22:32:30
Игорото: 1/5 (20%) — 27.08.2025, 22:23:50
Стас Пьеха: 5/5 (100%) — 27.08.2025, 21:37:20
Василий Петров: 2/5 (40%) — 27.08.2025, 21:36:07
Анишкова Анастасия: 3/5 (60%) — 27.08.2025, 21:30:21

[Скачать результаты \(CSV\)](#) [Очистить историю](#)

Рисунок 3. Результаты теста

	А	В	С	Д	Е	Ф	Г	Н	І
1	Имя	Правильн	Всего воп	Процент	Дата				
2	Анишкова	3	5	60%	27.08.2025, 21:30:21				
3	Василий Г	2	5	40%	27.08.2025, 21:36:07				
4	Стас Пыхов	5	5	100%	27.08.2025, 21:37:20				
5	иторото	1	5	20%	27.08.2025, 22:23:50				
6	Иванов И	5	5	100%	27.08.2025, 22:32:30				

Рисунок 4. Результаты теста в CSV-файле

Код инкапсулирует управление состоянием интерфейса через переключение CSS-классов, обеспечивая плавные переходы между этапами тестирования. Функции рендеринга вопросов, обработки выбора ответов и расчета результатов связаны в единый цикл работы приложения, что делает систему целостной и удобной для использования.

```
<script>
const questions = [
  {
    question: «1. Какое устройство является 'мозгом' компьютера?»,
    options: [«Оперативная память», «Процессор», «Жёсткий диск»,
«Монитор»],
    correct: 1
  },
  {
    question: «2. Что такое байт?»,
    options: [«8 бит», «1000 бит», «16 бит», «1024 бит»],
    correct: 0
  },
  {
    question: «3. Какой язык программирования используется для веб-
страниц?»,
    options: [«Python», «C++», «JavaScript», «Java»],
    correct: 2
  },
  {
    question: «4. Что такое IP-адрес?»,
```

```
options: [«Имя пользователя», «Физический адрес компьютера в
сети», «Адрес электронной почты», «Географический адрес»],
```

```
correct: 1
```

```
},
```

```
{
```

```
question: «5. Какой формат файла является векторной графикой?»,
```

```
options: [«JPEG», «PNG», «SVG», «GIF»],
```

```
correct: 2
```

```
}
```

```
];
```

```
// Элементы
```

```
const startScreen = document.getElementById(«startScreen»);
```

```
const quizContainer = document.getElementById(«quiz»);
```

```
const timer = document.getElementById(«timer»);
```

```
const timeDisplay = document.getElementById(«time»);
```

```
const submitBtn = document.getElementById(«submit»);
```

```
const restartBtn = document.getElementById(«restart»);
```

```
const resultDiv = document.getElementById(«result»);
```

```
const history = document.getElementById(«history»);
```

```
const historyList = document.getElementById(«historyList»);
```

```
let timerInterval;
```

```
let timeLeft = 5 * 60; // 5 минут
```

```
let studentName = «»;
```

```
// Загрузка при старте
```

```
window.onload = function () {
```

```
  loadHistory();
```

```
};
```

```
function startQuiz() {
```

```
  const input = document.getElementById(«studentName»);
```

```
  studentName = input.value.trim();
```

```
  if (!studentName) {
```

```
    alert(«Введите ваше имя!»);
```

```
    return;
```

```
  }
```

```
  startScreen.classList.add(«hidden»);
```

```
  quizContainer.classList.remove(«hidden»);
```

```
  timer.classList.remove(«hidden»);
```

```
  submitBtn.classList.remove(«hidden»);
```

```
updateTimerDisplay();
timerInterval = setInterval(updateTimer, 1000);

loadQuestions();
}

function loadQuestions() {
  quizContainer.innerHTML = «»;
  questions.forEach((q, index) => {
    const questionDiv = document.createElement(«div»);
    questionDiv.className = «question»;
    questionDiv.innerHTML = `
      <p>${q.question}</p>
      <div class=«options»>
        ${q.options.map((option, i) => `
          <div class=«option»>
            <input type=«radio» name=«question${index}» value=«${i}»
id=«q${index}opt${i}»>
            <label for=«q${index}opt${i}»>${option}</label>
          </div>
        `).join("")}
      </div>
    `;
    quizContainer.appendChild(questionDiv);
  });
}

function updateTimer() {
  timeLeft--;
  updateTimerDisplay();

  if (timeLeft <= 0) {
    clearInterval(timerInterval);
    submitQuiz();
    alert(«☐ Время вышло! Тест завершён.»);
  }
}

function updateTimerDisplay() {
  const minutes = Math.floor(timeLeft / 60);
  const seconds = timeLeft % 60;
  timeDisplay.textContent = `${minutes}:${seconds < 10 ? '0' :
"}${seconds}`;
}
```

```
function submitQuiz() {
  clearInterval(timerInterval);

  let correct = 0;
  let allAnswered = true;

  questions.forEach((q, index) => {
    const selected = document.querySelector(`input[name=«question${index}»]:checked`);
    if (!selected) {
      allAnswered = false;
    } else if (parseInt(selected.value) === q.correct) {
      correct++;
    }
  });

  if (!allAnswered && timeLeft > 0) {
    resultDiv.innerHTML = «✗ Пожалуйста, ответьте на все вопросы!»;
    resultDiv.className = «incorrect»;
    return;
  }

  const score = Math.round((correct / questions.length) * 100);
  let message;
  if (score === 100) {
    message = `🎉 ${studentName}, отлично! ${correct} из ${questions.length} правильных ответов!`;
  } else if (score >= 70) {
    message = `👍 ${studentName}, хорошо! ${correct} из ${questions.length} правильных.`;
  } else {
    message = `📉 ${studentName}, стоит повторить. Правильных: ${correct} из ${questions.length}.`;
  }

  resultDiv.innerHTML = message;
  resultDiv.className = score >= 70 ? «correct» : «incorrect»;

  // Сохранить результат
  saveResult({
    name: studentName,
    correct,
    total: questions.length,
    score,
```

```
        date: new Date().toLocaleString()
    });

    loadHistory();

    submitBtn.classList.add(«hidden»);
    restartBtn.classList.remove(«hidden»);
    history.classList.remove(«hidden»);
}

function saveResult(result) {
    const history = JSON.parse(localStorage.getItem(«quizHistory») || «[]»);
    history.push(result);
    localStorage.setItem(«quizHistory», JSON.stringify(history));
}

function loadHistory() {
    const historyData = JSON.parse(localStorage.getItem(«quizHistory») ||
«[]»);
    if (historyData.length === 0) {
        history.classList.add(«hidden»);
        return;
    }

    history.classList.remove(«hidden»);
    historyList.innerHTML = «»;
    const recent = historyData.slice(-5).reverse(); // последние 5
    recent.forEach(r => {
        const li = document.createElement(«li»);
        li.textContent = `${r.name}: ${r.correct}/${r.total} (${r.score}%) —
${r.date}`;
        historyList.appendChild(li);
    });
}

function exportToCSV() {
    const historyData = JSON.parse(localStorage.getItem(«quizHistory») ||
«[]»);

    if (historyData.length === 0) {
        alert(«Нет результатов для экспорта.»);
        return;
    }
}
```

```
const headers = [«Имя», «Правильные ответы», «Всего вопросов»,
«Процент», «Дата»];
const csvContent = [
  headers.join(«;»),
  ...historyData.map(r => [
    `»${r.name}»`,
    r.correct,
    r.total,
    `${r.score}%`,
    `»${r.date}»`
  ]).join(«;»)
].join(«\n»);

const blob = new Blob([«\uFEFF» + csvContent], { type:
«text/csv;charset=utf-8;» });
const url = URL.createObjectURL(blob);
const a = document.createElement(«a»);
a.href = url;
a.download      =      «результаты_теста_»      +      new
Date().toISOString().split(«T»)[0] + «.csv»;
a.style.display = «none»;
document.body.appendChild(a);
a.click();
document.body.removeChild(a);
URL.revokeObjectURL(url);
}

function clearHistory() {
  if (confirm(«Вы уверены, что хотите очистить всю историю?»)) {
    localStorage.removeItem(«quizHistory»);
    history.classList.add(«hidden»);
    alert(«История очищена.»);
  }
}

function restartQuiz() {
  timeLeft = 5 * 60;
  resultDiv.innerHTML = «»;
  resultDiv.className = «»;
  quizContainer.classList.add(«hidden»);
  timer.classList.add(«hidden»);
  submitBtn.classList.add(«hidden»);
  restartBtn.classList.add(«hidden»);
  history.classList.add(«hidden»);
}
```

```
startScreen.classList.remove(«hidden»);  
document.getElementById(«studentName»).value = «»;  
updateTimerDisplay();  
}  
</script>
```

4 Вывод

Разработанный интерактивный тест по информатике представляет собой полноценное веб-приложение с продуманным пользовательским интерфейсом и функциональной логикой. Благодаря адаптивному дизайну, интуитивной навигации, таймеру, системе оценивания и хранению истории результатов, приложение обеспечивает удобное и эффективное обучение. Реализация на HTML, CSS и JavaScript без использования внешних библиотек делает его легковесным и доступным для использования в образовательной среде.

Библиографический список

1. Бакулевская С. С. Подготовка педагогических работников к использованию технологий html5 для разработки электронных образовательных ресурсов // Информатика и образование. 2019. №. 5. С. 32-40.
2. Цилинская К.А. Создание и использование мультимедийных интерактивных плакатов по информатике //Будущее науки-2019. 2019. С. 340-344.
3. Королюк Я. А., Куликова Н. Ю. Опыт разработки и использования интерактивных образовательных ресурсов на основе html5 для онлайнобучения школьников информатике // XXV Всероссийская студенческая научно-практическая конференция Нижневартковского государственного университета. 2023. С. 48-55.
4. Размологов Д. М., Панина А. С., Захаров П. В. Использование html5 для разработки электронного учебника //Современные тенденции развития науки и технологий. 2016. №. 4-4. С. 92-94.
5. Шегаев И. С., Полозов А. Г. Новейший политологический глоссарий: основные категории, понятия, термины. Общество с ограниченной ответственностью Издательство Перо, 2015. С. 242-242.