

Рисование компьютерной мышью на холсте веб-страницы браузера с помощью инструмента Canvas

Кочитов Михаил Евгеньевич

*Приамурский государственный университет им. Шолом-Алейхема
студент*

Аннотация

В данной статье рассматривается рисование компьютерной мышью на холсте веб-страницы браузера с использованием языка программирования JavaScript и языка разметки HTML5, в котором добавлен инструмент Canvas, позволяющий рисовать в браузере. Также будет рассмотрен пример, где будет создан редактор рисования с инструментом кисти, палитры выбора цвета, выбор размера кисти и сам холст, на котором будет изображен собственный рисунок, используя компьютерную мышь.

Ключевые слова: HTML 5, JavaScript, веб-рисование, Canvas, холст, компьютерная мышь, веб-страница, браузер

Drawing with a computer mouse on canvas a browser web page using Canvas tool

Kochitov Mikhail Evgenevich

*Sholom-Aleichem Priamursky State University
student*

Abstract

This article discusses drawing with a computer mouse on canvas a browser web page using the JavaScript programming language and HTML5 markup language, in which the Canvas tool is added, which allows drawing in the browser. An example will also be considered where a drawing editor with a brush tool, a color picker, a choice of the size of the brush and the canvas itself will be created, on which your own drawing will be displayed using a computer mouse.

Keywords: HTML 5, JavaScript, web drawing, Canvas, canvas, computer mouse, web page, browser

Онлайн рисование и обработка фотографий в браузере довольно распространенная возможность для пользователей, которые избавляют их от установки необходимого программного обеспечения (графического редактора, например Photoshop, CorelDraw, GIMP и др.). Пользователям для использования онлайн-графического редактора понадобится всего лишь установленный браузер на устройстве, а также навыки рисования либо навыки обработки и улучшения качества изображений. В интернете большое количество веб-графических редакторов, и у каждого свои возможности

рисования. Инструмент, позволяющий рисовать в браузере, называется Canvas и он появился на свет в обновленном языке разметки HTML на версии 5. Язык программирования JavaScript позволит управлять самим веб-рисованием.

Целью данной статьи является рисование с помощью компьютерной мыши на холсте веб-страницы браузера с помощью технологии Canvas, используя язык программирования JavaScript и язык разметки HTML5. Также будет разработан пример веб-редактора с инструментом кисти, палитры выбора цветов, выбора размера кисти и сам холст, благодаря которому будет отображаться собственноручный рисунок.

В статье Гороховой Ю.А. рассматривается использование сервисов онлайн-рисования при обучении компьютерной графике на уроках информатики [1]. Рассматривая статью Кочитова М.Е. можно увидеть возможность веб-рисования на HTML5 с помощью инструмента Canvas [2]. Рылова А.В. в своей статье рассмотрела создание веб-приложения “Графики” [3]. Авторы Глухов Р.И. и Елиссев А.С. в своей статье рассмотрели разработку веб-редактора UML-моделей [4].

Для того, чтобы приступить к созданию своего веб-редактора рисования, то необходимо создать веб-страницу и написать в ней HTML код

```
<html>
  <head>
    <title>Веб-рисование в браузере</title>
  </head>
  <body>
    <h3>Веб-рисование в браузере</h3>
    <canvas id="paintBlock" style="border: 1px solid black"></canvas>
  </body>
</html>
```

Рисунок 1. Начальный HTML код веб-редактора

На рисунке 1 показан HTML код, который создает на веб-странице браузера холст, предназначенный для рисования, но он, однако, не имеет элементов рисования, таких как выбор цвета и размера. Элемент Canvas с идентификатором “paintBlock” будет использоваться в дальнейшем для управления веб-рисованием с помощью языка программирования JavaScript. В атрибуте “Style” указан стиль рамки холста толщиной в 1 пиксель черным цветом.

Далее рассмотрим следующий код, который будет использоваться для добавления стиля элементам управления веб-редактором, размещенным над холстом Canvas.

```
<style>
  .forColors {
    font-size: 40px;
    cursor: pointer;
  }
  .forSizes {
    font-size: 18px;
    cursor: pointer;
  }
</style>
```

Рисунок 2. CSS код добавления стилей элементов классов “forColors” и “forSizes”

На рисунке 2 представлен CSS код, который предназначен для создания стилей к элементам с классами “forColors” и “forSizes”. Параметр “font-size” задает размер шрифта в пикселях, а параметр “cursor” меняет изображение курсор на значок руки указательного пальца при наведении на элемент с определенными классом - “forColors” или “forSizes”.

Так как в CSS коде указаны два класса, которые не присвоены элементам, то добавим новый HTML код с помеченными для них классами

```
<div id="colors">Выбор цвета:<br>
  <span id="black" class="forColors" style="color: black">■</span>
  <span id="red" class="forColors" style="color: red">■</span>
  <span id="green" class="forColors" style="color: green">■</span>
  <span id="yellow" class="forColors" style="color: yellow">■</span>
  <span id="blue" class="forColors" style="color: blue">■</span>
  <span id="brown" class="forColors" style="color: brown">■</span>
  <span id="grey" class="forColors" style="color: grey">■</span>
  <span id="purple" class="forColors" style="color: purple">■</span>
  <span id="cyan" class="forColors" style="color: cyan">■</span>
  <span id="orange" class="forColors" style="color: orange">■</span>
</div>
<div id="sizes">Выбор размера:
  <span id="2" class="forSizes">Маленький</span>
  <span id="5" class="forSizes">Средний</span>
  <span id="10" class="forSizes">Большой</span>
</div>
<br>
<div>
  <button id="clear">Очистить</button>
</div>
<br>
```

Рисунок 3. HTML код элементов палитры выбора цветов и выбора размера кисти

Как видно на рисунке 3 изображен HTML код, в котором добавлены элементы с классами “forColors” для палитры выбора цветов и класса “forSizes” для выбора размера кисти. Элемент палитры цветов представляет из себя в одну строку последовательные “span” элементы, в которых написан символ квадрата и в атрибуте “style” указан в свойстве “color” определенный цвет, который также указан и в атрибуте “id”, являющийся идентификатором

каждого “span” элемента. Далее подобный блок кода написан и на выборе размера кисти в виде трех “span” элементов с соответствующими надписями и в атрибутах “id” указанным размером кисти в пикселях. Также имеется элемент кнопки “button” с id “clear” и надписью “Очистить”, предназначенный для полной очистки холста от всего, нарисованного кистью.

Далее необходимо запрограммировать веб-редактор, используя язык программирования JavaScript, в котором будут задействованы все HTML элементы

```
<script>
  var color = "black";
  var size = 10;
  var canvas = document.getElementById('paintBlock');
  var context = canvas.getContext("2d");
  canvas.width = 500;
  canvas.height = 500;
  canvas.addEventListener("mousemove", function(elem) {
    if (elem.which == 1) {
      context.fillStyle = color;
      context.fillRect(elem.offsetX, elem.offsetY, size, size);
    }
  });
  document.getElementById('colors').addEventListener("click", function(elem) {
    color = elem.target.id;
  });
  document.getElementById('sizes').addEventListener("click", function(elem) {
    size = elem.target.id;
  });
  document.getElementById('clear').addEventListener("click", function() {
    context.fillStyle = "white";
    context.fillRect(0, 0, 500, 500);
  });
</script>
```

Рисунок 4. JavaScript код веб-редактора

На рисунке 4, как можно заметить, представлен JavaScript код веб-редактора. Теперь рассмотрим этот код более подробно. Изначально объявляются глобальные переменные color (по умолчанию цвет кисти - черный) и size (по умолчанию размер кисти - большой или в 10 пикселей). Далее идет инициализация инструмента canvas с помощью функции “getContext” и задания ему указанного размера в 500 пикселей, используя функции “width” - ширина холста и “height” - высота холста. После добавляется зарегистрированная в JavaScript функция “addEventListener”, которая добавляет событие “mousemove”, которое срабатывает при движении компьютерной мыши по самому элементу canvas. В самой функции написана внутренняя функция, в которой добавлено условие “elem.which”, выдающее 1, когда нажата левая кнопка мыши. Далее при выполнении этого условия происходит рисование квадрата в том месте, где находится курсор, используя функции “fillStyle” - указание цвета кисти и функции “fillRect” - процесс рисования добавлением квадрата указанного размера в текущих координатах

месторасположения курсора компьютерной мыши. Иными словами, при удерживании левой кнопки мыши происходит рисование кистью там, где находится курсор мыши. Далее ниже указаны еще три функции “addEventListener” с указанием события “click”, которые выполняются по клику мыши (единоразовому нажатию левой кнопкой мыши). Первая функция привязана к элементу “colors”, которая присваивает глобальной переменной “color” новый цвет кисти через функцию “elem.target.id”, получающее значение идентификатора “span” элемента, по которому был произведен клик левой кнопки мыши. Тоже самое и сделано и для элемента “sizes”, который задает новый размер кисти. Последний элемент “clear” полностью очищает холст, в ее функции логичным образом очистка реализована нарисованием большого белого квадрата с размерами самого холста.

Далее рассмотрим в самом браузере реализованный веб-редактор

Веб-рисование в браузере

Выбор цвета:



Выбор размера: Маленький Средний Большой

Очистить

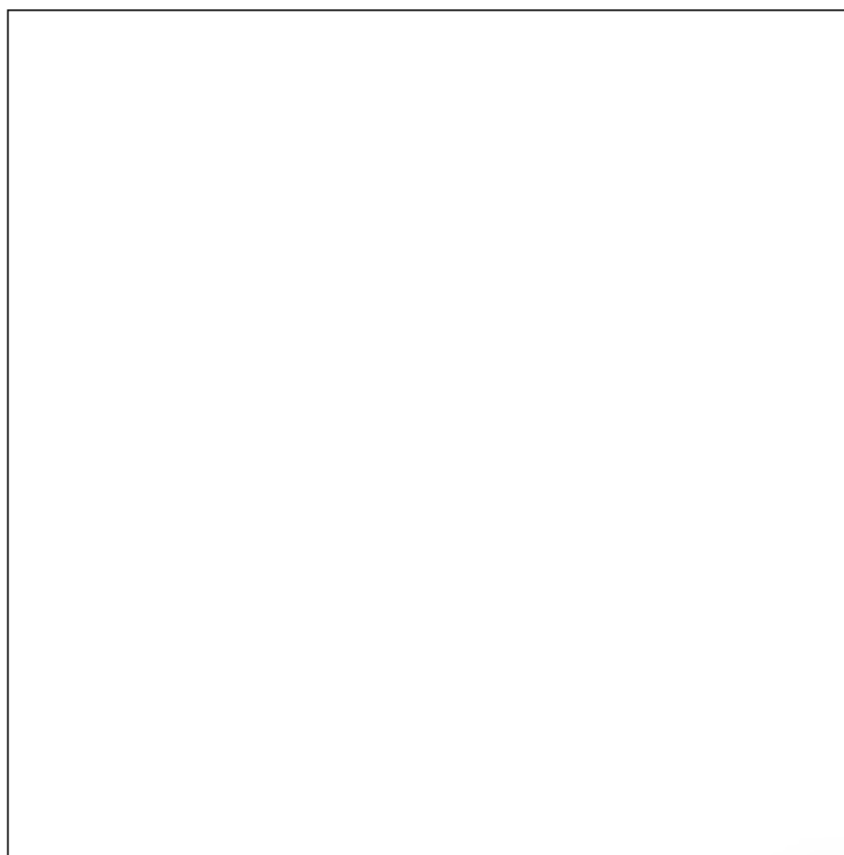


Рисунок 5. Готовый веб-редактор в браузере с пустым рисунком

На рисунке 5 изображен готовый веб-редактор на самой странице браузера, в котором видна палитра выбора цветов, выбор размера кисти и сам пустой холст, на котором будет непосредственно проводиться рисование различных детских рисунков (см. Рис. 6)

Веб-рисование в браузере

Выбор цвета:



Выбор размера: Маленький Средний Большой

Очистить



Рисунок 6. Готовый детский рисунок на холсте веб-редактора

Таким образом, была рассмотрена возможность рисования компьютерной мышью с использованием инструмента Canvas на языке разметки HTML5 и языке программирования JavaScript. Данный веб-редактор позволяет с помощью одного инструмента - кисти рисовать различные рисунки детского характера. Также данный веб-редактор можно улучшить, добавив новых инструментов, тем самым можно будет создавать рисунки уже в улучшенном виде. Онлайн графические редакторы необходимы каждому художнику и дизайнеру, чтобы свои художественные проекты создавать лишь имея установленный браузер и больше никакой другой программы.

Библиографический список

1. Горохова Ю.А. Использование сервисов онлайн-рисования при обучении компьютерной графике на уроках информатики // В сборнике: Современные проблемы и перспективы обучения математике, физике, информатике в школе и вузе Межвузовский сборник научно-методических трудов. Ответственный редактор С.Ф. Митенева. Вологда, 2019. С. 215-217.
2. Кочитов М.Е. Рассмотрение возможности веб-рисования на HTML5 с помощью Canvas // Постулат. 2018. № 7 (33). С. 17.
3. Рылова А.В. Создание веб-приложения “Графики” // Вестник современных исследований. 2019. № 2.13 (29). С. 42-45.
4. Глухов Р.И., Елисеев А.С. Разработка веб-редактора UML-моделей // Инновационные технологии: теория, инструменты, практика. 2015. Т. 1. С. 112-118.
5. Основы работы с Canvas в JavaScript. URL: <http://code.mu/books/javascript/canvas/osnovy-raboty-s-canvas-v-javascript.html> (дата обращения 20.08.2019)
6. JavaScript свойства и методы элемента Canvas. URL: <https://msiter.ru/references/canvas-reference> (дата обращения 20.08.2019)