УДК 681.3

ОСОБЕННОСТИ СОЗДАНИЯ АНИМАЦИИ В ВЕБ-ДОКУМЕНТАХ

Егорова И.Н., профессор, кафедра МСТ ХНУРЭ
Бережная С.А., студент, кафедра МСТ ХНУРЭ

Аннотация. *Исследованы методы и способы создания анимации в веб-документах с помощью двух наиболее востребованных технологий – HTML5 и CSS3. Приведены основные принципы использования анимации в веб-документах.*

Ключевые слова: *АНИМАЦИЯ, ТЕХНОЛОГИЯ, ОБЪЕКТ, ЭЛЕМЕНТ, КАДР, ГРАФИКА.*

Развитие веб-технологий способствует появлению анимации на сайтах. Она оживляет контент, путем внесения динамики в перегруженные текстом страницы. Анимация – основной тренд 2017 года в разработке сайтов. Элементы анимации в интерфейсах повышают интерес к использованию, делая их интуитивно понятными даже неопытному пользователю. Также ее присутствие помогает корпоративным сайтам наглядно продемонстрировать свой продукт и подчеркнуть все его положительные стороны [1]. Актуальность выбранной темы обусловлена тем фактом, что правильный выбор технологии создания анимации для веб-документов поможет в создании качественного контента.

Целью данной работы является исследование методов и способов создания простейших анимационных эффектов с помощью HTML5, CSS3 и определение целесообразности их использования.

Проектирование анимации позволило посетителям сайта интерактивно взаимодействовать с его интерфейсом и быстрее достигать своих целей. Чтобы добиться максимально эффективного результата от использования анимации в веб-документах, необходимо следовать 12 принципам анимации, которые создают у пользователей ощущение реальности.

1. Сплющивание и растяжение: создание ощущения наличия у элемента на странице физической массы.

2. Ожидание: реализация движения объектов в реалистичном виде. Достигается путем лёгкого покачивания, которое позволяет пользователям понять, что происходит.

3. Фокусировка: определение фокуса сцены, в то время как остальные элементы освобождают место для главного действия. В веб-документах это может быть реализовано как размещение контента на переднем плане с наложением темного фона, разной скорости движения объектов, подрагивания и мигания кнопки.

4. Движение «полный вперёд» и «пошаговое изменение». Прямое движение основано на полной отрисовке всех кадров анимации, а пошаговое – соответственно на создании последовательности ключевых кадров, в которых интервалы заполняются с помощью вспомогательных средств.

5. Следование и захлёстывание: указывает на остановку объекта. В случаях со списками управляемая графа игнорирует отведенное ей положение, а затем возвращается на нужную позицию.

6. Плавное ускорение и замедление: постепенное изменение скорости для естественного движения.

7. Дуги: реалистичное перемещение осуществляется только по кривой дуге, а не прямо.

8. Второстепенное действие: выделение того, на что пользователю необходимо обратить внимание при «разъезжающихся» компонентах (перетаскивание элемента для вставки в середину списка).

9. Время выполнения: длительность анимации выделяет ее на сайте среди контента.

10. Преувеличение: привлечение внимания к определённым действиям путем изменения размера элементов.

11. Объём: правило перспективы. При воспроизведении анимации изменение положения объектов на экране, браузер создает плавные переходы между состояниями.

12. Привлекательность: качественная и профессиональная подборка анимаций на одном сайте [2].

Веб-анимацию чаще всего реализуют с помощью графики в формате .gif и flash-анимации, а динамичность сайту придают java-скрипты. Но стоит уделить внимание HTML5 и CSS3, поскольку с их помощью также можно добиться отличных результатов.

С приходом HTML5 появился новый элемент Canvas. Благодаря ему интерактивные изображения, графики и диаграммы, игры, симуляции, любая другая анимация или приложения воспроизводятся в веб-браузере. Весь материал находится на веб-странице и теперь не нужно обращаться к Flash или иным приложениям [3].

Каждая анимация Canvas содержит набор простейших кадров. Для отображения на холсте одного кадра необходимо: очистить холст, сохранить, нарисовать объекты анимации, загрузить холст.

С помощью нового элемента HTML5 Canvas можно реализовать простую анимацию движения посредством функции setTimeout, которая рекурсивно вызывает саму себя после истечения отведенного периода времени и создает кадры движения. Дополнительно можно указать координаты перемещения.

Благодаря огромному множеству функций создается анимация без очистки экрана, таким образом, получается, что изображение будет плавно «прорисовываться». Больше всего это подходит для воплощения деловой или статистической графики.

При сложной анимации и играх разработчики прибегают к спрайтам. Спрайт – это файл изображения, хранящий в себе набор промежуточных кадров. Это

означает, что в одном изображении находится вся раскадровка анимации или состояний объекта. Дополнительно с помощью функций можно «оживить» спрайт.

Для вывода кадров спрайта используется метод контекста drawImage(img,sx,sy,swidth,sheight,x,y,width,height), где указаны все его параметры:

- img – вид графики;
- sx и sy – начальные координаты x и y обрезки;
- swidth и sheight – ширина и высота вырезаемого изображения;
- x и y – координаты размещаемого изображения на canvas;
- width и height – ширина и высота используемого изображения (растянуть или уменьшить изображение).

Вышеперечисленные методы являются основополагающими при разработке анимации в HTML5 Canvas. Эта технология имеет широкий спектр применения: от создания деловой графики до полноразмерных и масштабных браузерных игр [4].

Однако при разработке анимации с помощью элемента HTML5 Canvas можно столкнуться с рядом трудностей. Если возникает необходимость работы с текстом, то данная технология не в силах осуществить все задумки дизайнера. Дополнительные трудности может создать малый комплект примитивов. Нет дополнительных функций для работы с тенями. В целом отсутствует понятие графических объектов или групп операций, все возлагается на разработчика. Конечно, эти недочеты могут быть и спорными моментами, поскольку желаемый результат зависит от дизайна и самой структуры анимации.

Одним из способов оживить контент в HTML является использование, так называемых, CSS анимаций. С приходом CSS3 множество сложных задач стали простыми и решаемыми.

Свойство CSS transition устанавливает эффект перехода между двумя состояниями. Управлять можно с помощью свойств:

- transition-property – список свойств, предназначенных для анимирования;
- transition-duration – продолжительность анимации;
- transition-timing-function – контроль изменения скорости (точки ускорения и замедления). Для задания процесса анимации используются кривые Безье;
- transition-delay – задержка времени до начала анимации;
- transition – универсальное свойство, которое позволяет одновременно задать значения вышеперечисленных свойств.

CSS animations реализует более сложные анимации. Правило @keyframes создает анимацию с помощью набора ключевых кадров. Контролировать можно свойствами: animation-name – имя анимации; animation-iteration-count – количество повторов; animation-direction – задается направление анимации; animation-play-state – управление ее запуском и остановкой; animation-fill-mode – назначение свойств объекта до или после анимации.

MotionPathModule CSS позволяет создавать движение объектов по контуру через специальное свойство motion-path (поддержка только в Chrome). Ранее такую задачу можно было реализовать только с помощью SVG или сложных скриптов.

В данной спецификации можно управлять тремя свойствами: motion-path – указание точек(координат), по которым движется объект; motion-offset – движение элемента от начальной точки до конечной; motion-rotation – определение движущей стороны объекта [5].

Безусловно, анимации, реализованные с помощью технологии CSS3, наполняют сайт динамичностью, но не стоит забывать о возможных проблемах с производительностью, так как на изменение некоторых свойств требуется много ресурсов.

Интерактивная динамика на сайтах, созданная с помощью HTML5 и CSS3, очень распространена. Это обусловлено тем, что обе технологии являются официальными стандартами и поддерживаются всеми браузерами с 2014 года.

HTML5 и CSS3 обладают рядом весомых преимуществ. Эти технологии не требуют установки дополнительных плагинов на компьютер, они индексируются поисковыми системами, сайты намного «легче». Следует заметить, что применяется и JavaScript-анимация. Однако, последняя используется там, где не подходит CSS-анимация [6-8].

На основе проведенного анализа можно сделать вывод о целесообразности применения технологий создания анимации HTML5 и CSS3. Использование этих технологий позволяет добиться максимально эффективного результата применения анимации в веб-документах.

Литература.

1. Как оживить свой сайт. – Режим доступа: https://tproger.ru/digest/5-animation-tools-for-web/ – 28.03.2017. – Загл. с экрана.

2. Принципы анимации для веба. – Режим доступа: https://habrahabr.ru/company/htmlacademy/blog/255583/ – 28.03.2017. – Загл. с экрана.

3. Графика и Анимация в HTML5. – Режим доступа: http://www.incore.me/grafika-i-animaciya-v-html5/ – 30.03.2017. – Загл. с экрана.

4. Анимация встроенными средствами HTML5. – Режим доступа: http://x-site.by/info/html5-animation. – 30.03.2017. – Загл. с экрана.

5. Все об CSS3-анимации. – Режим доступа: https://stfalcon.com/ru/blog/post/animation-css. – 30.03.2017. – Загл. с экрана.

6. Анимация в сайте. – Режим доступа: http://hinex.ru/animacia-na-saite.php. – 30.03.2017. – Загл. с экрана.

7. Lyashenko, V. V., Matarneh, R., Baranova, V., & Deineko, Z. V. (2016). Hurst Exponent as a Part of Wavelet Decomposition Coefficients to Measure Long-term Memory Time Series Based on Multiresolution Analysis. American Journal of Systems and Software, 4(2), 51-56.

8. Кипень, Н. Ю., Бокарева, Ю. С., & Дейнеко, Ж. В. (2016). Исследование особенностей плоского и материал-дизайна в UI-интерфейсах.