

***ОСНОВЫ РАЗРАБОТКИ  
WEB-ИНТЕРФЕЙСОВ  
(2 СЕМЕСТР)***

**Анимация и спец-эффекты  
в вебе**

# *ПРОФЕССИОНАЛЬНАЯ ЗАДАЧА*

Реализовать спецэффекты и  
анимировать изображения  
на web-страницах

# ИСТОРИЯ АНИМАЦИИ В ВЕБЕ

История анимации насчитывает более 30 лет опыта использования разнообразных технологий, которые сменяли и улучшали друг друга

# *GIF*

15 июня 1987 года была опубликована первая «гифка», собственно, вот и она:



«Гифки» представляют собой последовательность статических картинок — в точности, как кадры в мультиках либо кинофильмах.

GIF поддерживает 256 цветов максимум плюс два байта на канал прозрачности.

Команда программистов в comuserve во главе со Стивом Уилхайтом — основателем формата — построила GIF вокруг чего-то совершенно непохожего на современную анимационную технологию.

Её основой стала компрессия лемпеля-зива-уэлча или LZW.

К тому времени, когда всемирная паутина начала активно набирать обороты, GIF стал монополистом.

В 1993 году появился передовой браузер mosaic, он поддерживал два формата изображения: проверенный GIF и новый JPEG.

Основное распространение «гифки» получили в рекламных баннерах. Анимированные объекты и сюжеты привлекали больше внимания потребителей к коммерческим предложениям, чем статичные изображения.

# *ADOBE FLASH PLAYER*

- Flash был создан американцем Джонатаном Гэйем ещё в 1996 году.
- Основанная им и его коллегами в 1993 году компания FutureWave Software планировала выйти на рынок графического ПО и занять на нём доминирующее положение.

SmartSketch (1993), FutureSplash Animator (1996) =>  
Macromedia Flash

# *ADOBE FLASH PLAYER*

- В 2005 году сама Macromedia была выкуплена компанией Adobe, а Macromedia Flash был в очередной раз переименован. На этот раз он стал называться **Adobe Flash**

1 декабря 2015 года Adobe объявила о переименовании Adobe Flash в **Adobe Animate** при следующем крупном обновлении. Это стало первым шагом по прекращению поддержки Adobe Flash Player. В Adobe Animate добавили экспорт в JavaScript (с использованием библиотеки CreateJS) с проигрыванием в Canvas.



# *ADOBE FLASH PLAYER*

- Летом 2017 года Adobe заявила, что остановит поддержку Flash 31 декабря 2020 года. Теперь, если зайти на официальный сайт компании, можно увидеть надпись на английском языке «Мы удалили Flash. Поддержка Flash закончилась».

Adobe Flash Player имел важное значение для творчества. С его помощью любой желающий даже без особых навыков программирования мог создать графические и звуковые шоу, выпустить собственную игру или сделать анимацию.

# *JAVA SCRIPT*

В 1995 году на помощь разработчикам пришёл JavaScript. Он помог снять ограничения, которые на разработчиков накладывали HTML и CSS.

Например, нужно всплывающее окно? Хочется динамически изменить порядок какой-то структуры? Ответ — JavaScript.

# CANVAS

Спецификация Canvas (как отдельной области на странице, внутри которой можно отображать графические объекты) была предложена в 2005 году компанией Apple для поддержки некоторых приложений внутри движка WebKit — на данный момент его используют браузеры Safari и Google Chrome.

Рабочая группа W3C включила Canvas в Web Applications 1.0, который вошёл в стандарт HTML 5.0.

**Canvas** — это стандартный элемент HTML5, он является всего лишь **контейнером для графики**. Чтобы нарисовать какой-нибудь графический объект, в действительности необходимо использовать специальный скрипт — обычно JavaScript.

# *CANVAS*

Canvas-анимации — отличный способ делать комплексные и красивые анимации. Взаимодействуя с пикселями, вы можете создавать сложные вещи и не терять в производительности. Это отличный выбор для сложных рисунков и взаимодействий.

# *CSS-АНИМАЦИИ*

Уже с середины 2000-х в W3C занимались включением анимации в CSS. Первая версия спецификации CSS-анимации увидела свет в 2009 году.

С помощью CSS-анимаций стали возможными переходы между исходным состоянием элемента и состоянием при наведении, нажатии или фокусе. Они также позволяют создавать более сложные эффекты, используя для этого наборы ключевых кадров.

# *CSS-АНИМАЦИИ*

С CSS-анимациями для решения большинства задач вам не понадобятся внешние библиотеки. У них отличная производительность.

Также их можно с лёгкостью применять в адаптивной разработке, ведь их легко модифицировать через использование медиа-запросов.

# *SVG*

SVG (**Scalable Vector Graphics**) — это двухмерная векторная графика, дословно «масштабируемая векторная графика».

В это же время SVG — текстовый формат, который можно легко править в блокноте или просто рисовать в векторных редакторах. Поэтому его можно использовать как изображение и в inline-формате.

Элементы в нём трактуются браузером как DOM-элементы, с ними можно работать так же, как с любыми другими DOM-элементами, в том числе анимировать с помощью CSS и JavaScript.

# *SVG*

В основу SVG легли языки разметки

VML (Vector Markup Language — язык векторной разметки) и

PGML (Precision Graphics Markup Language — язык разметки прецизионной графики).

Разрабатывается с 1999 года. В 2001 году появилась версия 1.0, а 10 годами позже, в 2011, вышла версия 1.1 — она до сих пор остаётся актуальной. Сейчас активно разрабатывается версия 2.0.



# *SVG*

В основу SVG легли языки разметки

VML (Vector Markup Language — язык векторной разметки) и

PGML (Precision Graphics Markup Language — язык разметки прецизионной графики).

У SVG-анимаций есть свой синтаксис — SMIL (The Synchronized Multimedia Integration Language). Он гораздо лучше подходит для создания SVG-анимаций, в отличие от CSS, потому что позволяет анимировать некоторые недоступные для CSS свойства.

# *WEBGL*

WebGL — Web Graphics Library (библиотека веб-графики). Первая спецификация была выпущена 3 марта 2011 года.

Эта технология в основном используется для сложных эффектов и 3D.

Также её можно использовать при создании анимаций для виртуальной реальности.

# *ИТОГ*

Сейчас для создания анимированных веб-страниц разработчики в основном используют CSS-анимации и переходы или нативный JavaScript. Но для более сложных анимаций, где не хватает возможностей CSS и JS, применяются SVG-анимации, Canvas-анимации и WebGL.

На данном курсе мы разберём базовые подходы и возможности CSS-анимаций и посмотрим их переключение нативным JavaScript.

# *ПРИМЕНЕНИЕ АНИМАЦИЙ В ВЕБЕ*

Любой интерфейс можно условно поделить на две составляющие.

Первая — непосредственно функционал, **что** он делает.

Вторая — детали, **как** он делает.

Внимание к деталям — это именно то, что выделяет хорошие сайты на общем фоне.

Микровзаимодействия

Обратная связь

Донесение информации о происходящем

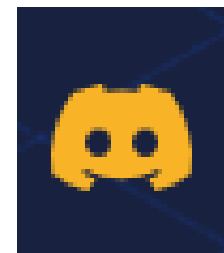
<https://codepen.io/aaroniker/pen/VdRRpM>



*ЗАДАВАЙТЕ ВОПРОСЫ ПО  
ЛЕКЦИИ!*



**канал в дискорд**



**[https://discord.gg/  
wKft5yhH47](https://discord.gg/wKft5yhH47)**