

УДК 004.588

ВАЖНОСТЬ ОСНОВНЫХ CSS-ТЕХНОЛОГИЙ НА ПРИМЕРЕ РАЗРАБОТЫВАЕМОГО ИГРОВОГО ВЕБ-ПРИЛОЖЕНИЯ «ОБУЧЕНИЕ ВОЗМОЖНОСТЯМ CSS»

Р.П. ГУСЕВСКИЙ

(Представлено: канд. тех. наук, доц. И.Б. БУРАЧЕНОК)

Приводится обоснование необходимости в разработке современных игровых приложений. Анализируются используемые технологии при создании веб-приложений. Подробно рассмотрены CSS-технологии, с целью их использования для демонстрации в виде увлекательной игры «Обучение возможностям CSS»

Большинство людей проводят свое свободное время в играх, особенно дети и подростки. Есть такой подход к обучению, как подача материала для изучения в игровой форме, которая будет полезна и интересна людям любого возраста. В рамках данной темы ведется разработка кроссплатформенного, игрового веб-приложения «Обучение возможностям CSS».

Рассмотрим, что такое CSS. Это Каскадные таблицы стилей, которые являются следствием дальнейшего развития HTML и дают нам возможность перейти на следующий уровень представления информации. Таблицы стилей позволяют разделить смысловое содержимое странички и его оформление. При изучении CSS можно столкнуться с рядом проблем: с различными стилями, свойствами, а также с технологиями для верстки блоков и таблиц. Что можно сказать о HTML – это язык разметки, который является инструментом для создания гиперссылок, вставки изображений, таблиц и много другого [1].

В современном мире веб-технологии и языки программирования развиваются очень быстрыми темпами, соответственно увеличивается и количество веб-разработчиков, а также желающих пользователей, которые хотят изучать или развивать свои навыки программирования. Однако при изучении веб-технологий многие сталкиваются с различными проблемами, и тратят много времени на поиски решений той или иной проблемы. Для того, чтобы облегчить изучение CSS, предлагается разработать веб-приложение, которое ориентировано на обучение сложным свойствам CSS и его технологиям верстки: flexbox, grid layout, работа с анимацией, позиционирование элементов на веб-странице при помощи свойства position. Благодаря визуализации и игровой подаче, рассматриваемые технологии станут доступнее для изучения и запоминания материала. Вряд ли можно стать программистом только играя. Но можно получить положительный заряд на старте, изучить основные алгоритмические конструкции и логику программирования, приобрести навыки планирования действий при решении практических задач [2].

При реализации данного веб-приложения планируется использование следующих технологий: ReactJS – в качестве фреймворка на клиентской стороне, для разработки серверной части – программная платформа Node.js и фреймворк Express, а также язык программирования Javascript. ReactJS – JavaScript-библиотека с открытым исходным кодом для разработки пользовательских интерфейсов. React может использоваться для разработки одностраничных и мобильных приложений. Его цель – предоставить высокую скорость, простоту и масштабируемость. В качестве библиотеки для разработки пользовательских интерфейсов React часто используется с другими библиотеками, такими как Redux [3]. Node.js – программная платформа, основанная на движке V8 (транслирующем JavaScript в машинный код), превращающая JavaScript из узкоспециализированного языка в язык общего назначения [4]. Express – это минималистичный и гибкий веб-фреймворк для приложений Node.js, предоставляющий обширный набор функций для мобильных и веб-приложений [5]. JavaScript – это полноценный динамический язык программирования, который применяется к HTML документу, и может обеспечить динамическую интерактивность на веб-сайтах [6].

Приложение будет состоять из двух страниц: основная страница с перечисленными пунктами изучаемых технологий и различных свойств CSS и динамически подгружаемая страница с самой игрой, которая характеризует выбранную технологию.

Далее осуществим описание технологий верстки и свойств, которые используются в приложении и с которыми чаще всего у людей возникают проблемы. В CSS имеется 4 самых проблемных технологии: flexbox, grid layout, работа с анимацией, позиционирование элементов на веб-странице при помощи свойства position.

Flexbox – это новый способ располагать блоки на странице. Это технология, созданная именно для раскладки элементов, в отличие от float-ов. С помощью Flexbox можно легко выравнивать элементы по горизонтали и по вертикали, менять направление и порядок отображения элементов, растягивать блоки

на всю высоту родителя или прибавить их к нижнему краю [7]. Но есть один нюанс, что эта технология поддерживается не всеми браузерами:

- 1) IE: 11.0, 10.0 -ms-;
- 2) Firefox: 28.0, 18.0 -moz-;
- 3) Chrome: 29.0, 21.0 -webkit-;
- 4) Safari: 6.1 -webkit-;
- 5) Opera: 12.10 -webkit-;
- 6) iOS Safari: 6.1 -webkit-;
- 7) Opera Mini: 8;
- 8) Android Browser: 4.4, 4.1 -webkit-;
- 9) Chrome for Android: 44.

Проблемы с данной технологией возникают тогда, когда огромную роль играет поддержка старых браузеров, в таком случае нужно отказаться от данной технологии. Однако на данный момент развитие идет в сторону использования новых технологий нежели поддержки старых браузеров. Следует отметить, что многие пользователи не знают какие свойства существуют и как ими пользоваться для правильного расположения элементов. Имеется как старый синтаксис, так и новый, и у каждого из них имеются свои нюансы и с этим тоже необходимо разобраться. Таким образом, в разрабатываемом приложении обязательно будут рассмотрены все перечисленные аспекты.

Следующей технологией для рассмотрения была выбрана технология Grid layout. Grid layout – приносит инструмент двумерного макета в веб, с возможностью размещения элементов в строках и столбцах. CSS-сетка может использоваться для достижения многих различных макетов. Он разделяет страницу на крупные регионы, или определяет отношения с точки зрения размера, положения и слоя, между частями элемента управления, построенных из HTML-примитивов.

Как и таблицы, Grid layout дает возможность автору позиционировать элементы в колонках (columns) и строках (rows). Тем не менее, в отличие от таблиц, Grid layout не имеет контентной структуры, поэтому включение в таблицы большого количества макетов невозможно [8].

CSS3-анимация придает сайтам динамичность. Она оживляет веб-страницы, улучшая взаимодействие с пользователем. В отличие от CSS3-переходов, создание анимации базируется на ключевых кадрах, которые позволяют автоматически воспроизводить и повторять эффекты на протяжении заданного времени, а также останавливать анимацию внутри цикла.

CSS3-анимация может применяться практически для всех html-элементов, а также для псевдо-элементов :before и :after. При создании анимации не стоит забывать о возможных проблемах с производительностью, так как на изменение некоторых свойств требуется много ресурсов [9].

Далее рассмотрим свойство CSS position, которое устанавливает способ позиционирования элемента относительно окна браузера или других объектов на веб-странице. Данное свойство принимает 4 основных значения:

- 1) Absolute – указывает, что элемент абсолютно позиционирован, при этом другие элементы отображаются на веб-странице словно абсолютно позиционированного элемента и нет. Положение элемента задается свойствами left, top, right и bottom, также на положение влияет значение свойства position родительского элемента. Так, если у родителя значение position установлено как static или родителя нет, то отсчет координат ведется от края окна браузера. Если у родителя значение position задано как fixed, relative или absolute, то отсчет координат ведется от края родительского элемента;
- 2) Fixed – по своему действию это значение близко к absolute, но в отличие от него привязывается к указанной свойствами left, top, right и bottom к точке на экране и не меняет своего положения при прокрутке веб-страницы.
- 3) Relative – положение элемента устанавливается относительно его исходного места. Добавление свойств left, top, right и bottom изменяет позицию элемента и сдвигает его в ту или иную сторону от первоначального расположения;
- 4) Static – элементы отображаются как обычно. Использование свойств left, top, right и bottom не приводит к каким-либо результатам.
- 5) Inherit – наследует значение родителя [10].

Таким образом, вышеописанные технологии и свойства CSS изначально будут включены в разрабатываемое веб-приложение с целью исключить слабые места разработчиков, а также ознакомить и дать правильную базу знаний пользователям для работы с этими технологиями.

Заключение. Все описанные в статье нюансы, будут учтены в разрабатываемом веб-приложении. В перспективе планируются доработки данного приложения с помощью добавления новыми технологиями CSS.

ЛИТЕРАТУРА

1. Web2Studio – Яркие решения [Электронный ресурс]. – 2019. – Режим доступа: – <http://webstudio2u.net/ru/programming/127-programming.html>. – Дата доступа: 07.09.2019
2. Игры, которые учат программированию / Хабрахабр [Электронный ресурс]. – 2019. – Режим доступа: – <https://m.habrahabr.ru/post/273003>. – Дата доступа: 07.09.2019.
3. ReactJS / Википедия [Электронный ресурс]. – 2019. – Режим доступа: – <https://ru.wikipedia.org/wiki/React>. – Дата доступа: 08.09.2019.
4. NodeJS / Википедия [Электронный ресурс]. – 2019. – Режим доступа: – <https://ru.wikipedia.org/wiki/Node.js>. – Дата доступа: 08.09.2019.
5. Express – фреймворк веб-приложений Node.js / [Электронный ресурс]. – 2019. – Режим доступа: – <http://expressjs.com/ru>. – Дата доступа: 08.09.2019.
6. Основы Javascript / MDN [Электронный ресурс]. – 2019. – Режим доступа: – https://developer.mozilla.org/ru/docs/Learn/Getting_started_with_the_web/JavaScript_basics. – Дата доступа: 08.09.2019.
7. CSS 3 flexbox – описание всех свойств модуля, примеры верстки / [Электронный ресурс]. – 2019. – Режим доступа: – <https://html5book.ru/css3-flexbox>. – Дата доступа: 12.09.2019.
8. CSS Grid Layout / MDN [Электронный ресурс]. – 2019. – Режим доступа: – https://developer.mozilla.org/ru/docs/Web/CSS/CSS_Grid_Layout. – Дата доступа: 13.09.2019.
9. CSS3-анимация (свойство animation) [Электронный ресурс]. – 2019. – Режим доступа: – <https://html5book.ru/css3-animation>. – Дата доступа: 14.09.2019.
10. Position / htmlbook.ru [Электронный ресурс]. – 2019. – Режим доступа: – <http://htmlbook.ru/css/position>. – Дата доступа: 14.09.2019.