ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ КОМПОНЕНТНОГО ПОДХОДА ПРИ РАЗРАБОТКЕ ВЕБ-САЙТОВ

***Егорова И.Н.**, к.т.н., проф., кафедра МСТ ХНУРЭ*
***Худолей А.Ю.**, магистрант, кафедра МСТ ХНУРЭ*

Процесс создания веб-приложений подвержен постоянным изменениям. Несмотря на то, что веб-технологии стремительно развиваются: создаются новые фреймворки и библиотеки, позволяющие ускорить процесс разработки, а также упростить поддержку приложений, все еще остается ряд проблем, наиболее важными из которых являются: слабая семантика, конфликты стилей, отсутствие шаблонов и отсутствие стандартов.

Просмотрев код любой веб-страницы, легко обнаружить, что в основном он состоит из элементов. Если не учитывать название классов и идентификаторов, то сложно понять какой смысл несет каждый элемент – в этом и заключается слабая семантика (рис. 1)

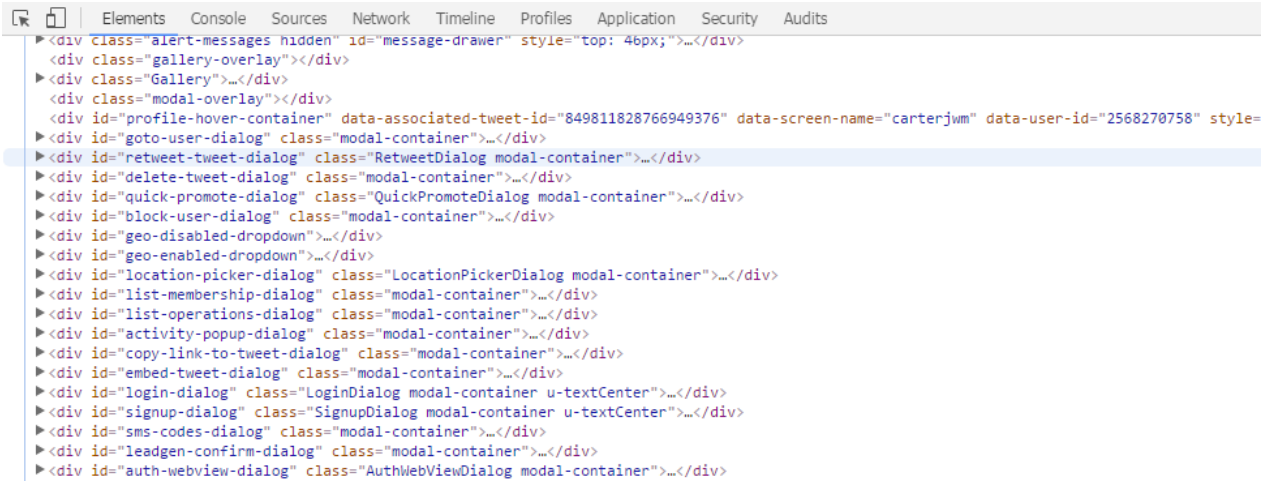

Рисунок 1 – Пример HTML кода

Проблема конфликтов стилей возникает, когда в разработке веб-сайта участвует команда из нескольких человек, и каждый из разработчиков имеет возможность добавить или изменить стили любого элемента страницы. В таком случае нет гарантии, что изменения не повлияют на другие элементы.

Отсутствие шаблонов при разработке веб-сайтов также имеет свои недостатки. Например, необходимо вывести на странице список новостей, каждая из которых имеет заголовок, дату, текст и автора. Если исключить использование каких-либо инструментов для работы с шаблонами, то создается скрытый блок, который будет дублироваться и наполняться необходимой информацией. Хотя изначально блок скрыт, но в DOM дереве он присутствует, а следовательно увеличивается время загрузки страницы.

Для написания отдельных элементов и пользовательских компонентов нет единых правил. В различных библиотеках и фреймворках свои правила и условия. Эту проблему и все вышеперечисленные решает новый комплекс

стандартов – веб-компоненты. Эти стандарты позволяют создавать и многократно использовать компоненты в веб-приложениях. Их использование не требует дополнительных зависимостей и изучения новых библиотек.

Веб-компоненты разработаны Консорциумом Всемирной паутины (W3C). Они позволяют разработчикам не только создавать, но и эффективно описывать уже существующие реализации HTML элементов. Веб-компоненты – совокупность спецификаций Custom Elements, Shadow DOM, HTML Templates и HTML Imports [1]. Каждая из спецификаций независима и ее можно использовать не только как отдельную единицу, но и в совокупности с другими.

Custom Elements решает проблему слабой семантики. Эта спецификация позволяет разработчикам не только расширять уже существующие, но и создавать свои собственные HTML-теги, а затем использовать их как любые стандартные теги, описывая для них свои свойства и методы. Custom Elements уже имеют 2 редакции спецификации v0 и v1. В последней редакции немного изменен жизненный цикл элемента, т.е. его создание, добавление и т.п.

Shadow DOM решает проблему конфликтов стилей, позволяя инкапсулировать стилизацию и внутреннюю разметку компонента, скрывая детали реализации будущего компонента от конечного пользователя. Shadow DOM имеет также 2 редакции спецификации. В первой спецификации у Shadow DOM был только открытый режим, который позволял получить доступ к элементам при помощи JavaScript. Во второй редакции был добавлен еще и закрытый режим, который ограничивает доступ к элементам. Одной из особенностей Shadow DOM является связь с Light DOM, что позволяет непосредственно из Shadow DOM обратиться ко всем дочерним узлам элемента, которые находятся вне этого веб-компонента и отобразить их в любом его месте [2].

HTML Templates решает проблему отсутствия шаблонов, определяя шаблон для нового элемента. Этот шаблон позволяет хранить некоторую разметку для страницы, которую впоследствии можно клонировать и повторно использовать. Изначально шаблон скрыт и не является частью DOM, он проверяется на валидность парсером, но только при его инициализации содержимое шаблона выводится. HTML Templates имеет самую стабильную спецификацию.

Проведенное исследование позволяет заключить, что, несмотря на ограниченную поддержку браузерами этих технологий, веб-компоненты могут быть полезны при разработке веб-сайтов. Их использование позволяет решить ряд значительных проблем и сделать приложения более гибкими и легко расширяемыми, без каких-либо сторонних фреймворков и библиотек.

Список литературы

1. Overson, J. Developing Web Components / J. Overson, J. Strimpel. – O'Reilly Media, Inc., 2015. – P. 5-21.

2. Современный учебник JavaScript. – Режим доступа: https://learn.javascript.ru/ webcomponents. – 23.04.2017. – Загл. с экрана.

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