

УДК 004.05

Толстых М.А.
магистрант, УО БГУИР,
г. Минск, Беларусь

АДАПТИВНАЯ ЗАГРУЗКА WEB - РЕСУРСОВ

Аннотация

Данная статья посвящена анализу способов улучшения скорости загрузки web - ресурсов для слабых устройств.

Ключевые слова

Web - ресурсы, web - технологии, адаптивная загрузка, программирование.

Возможности устройств и сетевые подключения сильно различаются. Сайты, которые нравятся пользователям на устройствах высокого класса, могут быть непригодны для использования на устройствах низкого уровня. Любой пользователь может столкнуться с

медленным сайтом, поэтому разработка универсальных решений может не всегда работать [1].

В своем выступлении на саммите Chrome Dev Summit Адди Османи из Google и Нейт Шлосс из Facebook исследуют решение этой проблемы - паттерны для доставки страниц, которые лучше подходят для различных ограничений со стороны пользовательских устройств и сетей. Они называют это адаптивной загрузкой [2].

Адаптивная загрузка предполагает предоставление разным пользователям уникальный опыт в зависимости от их сетевых и аппаратных ограничений:

- быстрое ядро для всех пользователей (включая устройства начального уровня);
- постепенное добавление высококачественных функций, если сеть и оборудование пользователя могут с этим справиться.

Оптимизируя работу с учетом конкретных аппаратных и сетевых ограничений, вы даете возможность каждому пользователю получить максимальную отдачу от своего устройства [3]. Адаптация опыта к ограничениям пользователей может включать:

- загрузка изображений и видео низкого качества в медленных сетях;
- регулировка частоты кадров анимации на недорогих устройствах;
- избегание дорогостоящих в вычислительном отношении операций на устройствах низкого уровня;
- блокировка сторонних скриптов на более медленных устройствах;
- загрузка некритического JavaScript для интерактивности только на быстрых процессорах.

Индикаторы, которые вы можете использовать для адаптивной загрузки:

- Сеть - для точной настройки передачи данных с целью использования меньшей полосы пропускания (через `navigator.connection.effectiveType`). Вы также можете использовать пользовательские настройки Data Saver (через `navigator.connection.saveData`);
- Память - для уменьшения потребления памяти на устройствах низкого уровня (через `navigator.deviceMemory`);
- Количество ядер ЦП - для ограничения дорогостоящего выполнения JavaScript и уменьшения логики, интенсивно использующей ЦП, когда устройство не может справиться с этим должным образом (через `navigator.hardwareConcurrency`).

Есть два места, где вы можете принять решение о том, что обслуживать пользователей: клиент и сервер. На клиенте у вас есть указанные выше API JavaScript. На сервере вы можете использовать клиентские заголовки, чтобы получить представление о возможностях устройства пользователя и сети, к которой он подключен.

Адаптивная загрузка — это все о проектировании ваших сайтов с учетом инклюзивности. Создайте базовый интерфейс, который отлично подходит для всех, а затем переключайте или накладывайте функции, которые делают его еще более привлекательным, если у пользователя достаточно памяти, процессора или быстрой сети [4].

Список использованных источников и литературы:

1. The cost of JavaScript [Электронный ресурс]. URL: <https://v8.dev/blog/cost-of-javascript-2019> (Дата обращения 24.12.2020).

2. Adaptive Loading [Электронный ресурс]. URL: <https://www.youtube.com/watch?v=puUPpVrIRkc> (Дата обращения 24.12.2020).

3. Adapting for users [Электронный ресурс]. URL: <https://developers.google.com/web/fundamentals/performance/optimizing-content-efficiency/client-hints> (Дата обращения 24.12.2020).

4. GITHUB Demos [Электронный ресурс]. URL: <https://github.com/GoogleChromeLabs/adaptive-loading#full-applications> (Дата обращения 25.12.2020).

© М.А. Толстых, 2021