

Цели занятия

- узнать основные термины HTTP;
- узнать что такое NodeJS и NPM и установить к себе на компьютер;
- узнать что такое React и создать пустой проект по шаблону;

Полезные материалы

- [Простым языком про HTTP](#)
- [Официальный сайт NodeJS](#)
- [Документация по React](#)
- [CreateReactApp — создать новый проект одной командой](#)

Что такое HTTP?

Мы начинаем модуль, который посвящен разработке сайтов с помощью React, но прежде чем приступить к его изучению нам необходимо поверхностно познакомиться с некоторыми технологиями и инструментами.

HTTP — это протокол (HyperText Transfer Protocol), позволяющий получать различные ресурсы, например HTML-документы. Протокол HTTP лежит в основе обмена данными в Интернете. HTTP является протоколом клиент-серверного взаимодействия, что означает инициирование запросов к серверу самим получателем, обычно веб-браузером (web-browser). Полученный итоговый документ будет (может) состоять из различных поддокументов, являющихся частью итогового документа: например, из отдельно полученного текста, описания структуры документа, изображений, видео-файлов, скриптов и многого другого.

Именно поэтому почти все ссылки в интернете начинаются с `http://`, так как это и есть указание протокола передачи данных. В современном интернете чаще всего используется `https://`, где буква `s` означает `secure`, т.е безопасный.

На самом деле существует несколько протоколов, каждый из которых создан для своей цели и их изучение выходит за рамки данного курса. Например, вы могли видеть ссылки, которые начинаются с `ftp://`

Что такое хост и домен?

Хост (от англ. `host` – «хозяин, принимающий гостей») – какое-либо устройство, предоставляющее услуги по технологии «клиент - сервер» по каким-либо интерфейсам и уникально определенное на этом сервере.

Это то место, где хранится различная информация, например, ваш сайт, потому что понятие «хост» чаще всего используется в связи с размещением веб-ресурса в интернете. В более частном случае хостом называется локальный компьютер, подключенный к интернету (`localhost`). Для обозначения хоста используется его сетевое имя – если хостом является домашний компьютер, и домен или `ip`-адрес – если речь идет о хосте в интернете.

Домен (от англ. `domain` – «сфера», «территория») – это область пространства иерархических имен, идентифицирующихся доменом. Иначе говоря, под доменом понимается адрес сайта, его имя. Имя сайту, или адрес, необходимо затем, чтобы указать посетителям дорогу к ресурсу.

Каждый входящий в интернет компьютер имеет свой уникальный доменный адрес (`domain address`), или по-другому доменное имя (`domain name`), или просто имя узла (`host name`). Адрес, или домен, существует в виде слова, иногда цифр, разделенные точками. От количества точек в имени зависит, домен какого уровня перед нами.

Что такое NodeJS?

NodeJS – это серверная реализация языка программирования JavaScript. Грубо говоря, это тот же JavaScript, только работающий за пределами браузера. Изначально JavaScript создавался исключительно как браузерный язык, с помощью него добавляли интерактивности в веб-страницы. В 2014 году Райан Даль создал Node.js и тем самым расширил возможности языка программирования JavaScript.

В современном мире NodeJS используется всё чаще, что даёт вам возможность изучив JavaScript в браузере довольно быстро начать писать JavaScript на сервере и заниматься бэкенд-разработкой.

Он бесплатный и его можно скачать и установить с [официального сайта](#) (выбирайте версию с пометкой LTS), а также NodeJS внутри себя уже имеет NPM...

Что такое NPM?

NPM используют программисты для установки готовых решений в свои проекты. В мире PHP и WordPress существуют плагины, которые можно добавлять в свой сайт расширяя его базовый функционал. Вот NPM – это почти то же самое, только для JavaScript и NodeJS.

Знающие люди могут отметить, что пример с плагинами на WordPress некорректный, ведь у языка программирования PHP есть свой аналог NPM. В любом случае это было сделано нами лишь для упрощения усвоения материала и многие современные языки также имеют свои менеджеры проектов: у языка программирования PHP – Composer, у Python – Pip, у C# – NuGet, а у Ruby – RubyGems.

И на самом деле создатели некоторых из них вдохновлялись NPM и не скрывают этого. Также, стоит отметить, что NPM содержит в себе настолько много готовых пакетов, что если сложить все пакеты из RubyGems, NuGet и Composer, то там все равно будет больше. Это все связано с популярностью JavaScript, ведь это один из самых популярных языков в мире.

Как работать с NPM?

После того как вы установили себе на компьютер NodeJS, у вас также будет установлен и NPM и вы можете выполнять некоторые его команды в терминале.

Чаще всего вы будете использовать его для установки каких-то сторонних пакетов с помощью команды `npm install` название-пакета, после чего NPM скачает данный пакет и вы сможете его использовать.

Обратите внимание, что в этот момент NPM создаст директорию `node_modules` и её НЕ НУЖНО хранить в репозитории. А также будут созданы два файла `package.json` и `package-lock.json`, которые обязательно нужно хранить в Git и которые не нужно редактировать вручную.

Когда вы устанавливаете или удаляете какой-то пакет, то NPM делает об этом запись в `package.json` и сохраняет копию пакета в `node_modules`. Вы всегда можете удалить `node_modules` и заново скачать все нужные пакеты с помощью команды `npm install`, если у вас есть файл `package.json`

Что такое React?

React — это библиотека JavaScript с открытым исходным кодом для создания внешних пользовательских интерфейсов.

React ориентируется на создании представлений приложений через инкапсулированные единицы (называются компонентами), которые сохраняют состояние и генерируют элементы пользовательского интерфейса. Вы можете комбинировать и переиспользовать компоненты как вам угодно, а также использовать компоненты других разработчиков, просто устанавливая их с помощью NPM.

Более подробно мы будем изучать возможности React в последующих занятиях данного модуля, где разберем основные вещи из которых состоит React и научимся применять его для упрощения разработки сайта.

Что такое Virtual DOM?

DOM расшифровывается как Document Object Model (объектная модель документа). Проще говоря, DOM — это представление пользовательского интерфейса (user interface, UI) в приложении. При каждом изменении UI, DOM также обновляется для отображения этих изменений. Частые манипуляции с DOM негативно влияют на производительность.

DOM представляет собой древовидную структуру данных. Поэтому изменения и обновления самого DOM являются достаточно быстрыми. Но после изменения обновленный элемент и все его потомки (дочерние элементы) должны быть повторно отрисованы (отрендерены) для обновления UI приложения. Повторный рендеринг — очень медленный процесс. Таким образом, чем больше у нас компонентов UI, тем более дорогими с точки зрения производительности являются обновления DOM.

В React для каждого объекта настоящего DOM существует соответствующий объект Virtual DOM. VDOM — это объектное

представление реального DOM, его легковесная копия. VDOM содержит те же свойства, что и реальный DOM, но не может напрямую влиять на то, что отображается на экране.

Когда в UI добавляются новые элементы, создается VDOM в виде дерева. Каждый элемент является узлом этого дерева. При изменении состояния любого элемента, создается новое дерево. Затем это новое дерево сравнивается (diffed) со старым.

После этого вычисляется наиболее эффективный метод внесения изменений в реальный DOM. Цель данных вычислений состоит в минимизации количества операций, совершаемых с реальным DOM. Тем самым, уменьшаются накладные расходы, связанные повторным рендером и поэтому сайты, сделанные на React имеют очень хорошую производительность, даже если на них происходит много различных изменений.