[**Тесты**](https://proglib.io/p/category/tests/)

React Developer и путь его развития в 2019 году

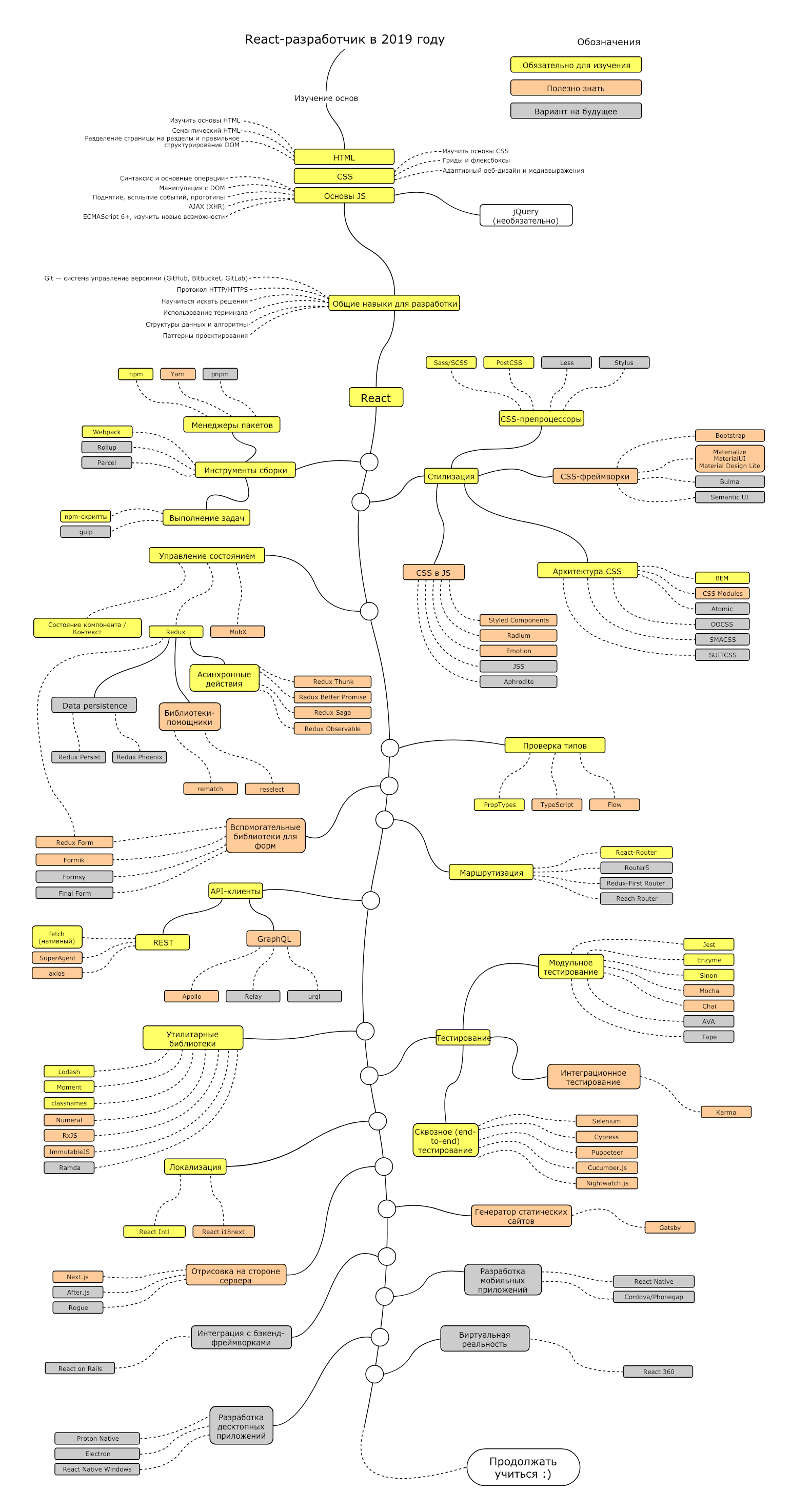
Направление «React Developer» в прошлом году достигло пика популярности и не намерено снижать темпы. Так давайте же рассмотрим путь React-развития.

Самыми популярными JS-фреймворками на сегодняшний день являются React, Angular и Vue. По данным, собранным сайтом StackOverflow, React выбирают 67%, а Angular – 42% пользователей. У каждого продукта есть слабые и сильные стороны, но React – действительно более правильный выбор:

* имеет низкий порог вхождения;
* простой переход с другого фреймворка;
* элементарное обновление, обратная совместимость и переход на другую версию (в Angular, например, «напрямую» не получится перейти с 2.0 на 3.0);
* идеальный выбор для крупных проектов.

React – JavaScript-библиотека, используемая в Facebook и Instagram. Эти два гиганта активно участвуют в улучшении библиотеки, поэтому интерес к ней еще долго не угаснет. Высокая производительность продуктов, написанных на React, достигается с использованием Virtual DOM, удобство использования гарантируется изоморфностью, ну и про повторное использование кода в различных проектах не стоит забывать.

React Developer начинает движение



Отправные точки в развитии

А теперь подробнее разберем ключевые элементы схемы, и что необходимо изучить на каждом из этапов.

I. Основы

1. HTML

* изучить основы HTML;
* сделать несколько страниц в качестве упражнения.

1. CSS

* создать страницу с использованием сеток и флексбокса;
* стилизовать страницы.

1. Основы JS

* познакомиться с синтаксисом;
* изучить основные DOM-операции;
* изучить JS-механизмы (Hoisting, Event Bubbling, Prototyping);
* потренировать AJAX-вызовы (XHR);
* изучить новые возможности ECMAScript 6+;
* вспомнить / познакомиться с jQuery.

II. Общие навыки для разработки

* создать несколько репозиториев, расшарить свой код;
* изучить HTTP(S) и методы (GET, POST, PUT, PATCH, DELETE, OPTIONS);
* научиться пользоваться [поиском Google](http://www.powersearchingwithgoogle.com/);
* поюзать терминал и изучить оболочку (bash, zsh, fish);
* почитать про алгоритмы, структуры данных и паттерны проектирования;

III. Изучить [официальный сайт](https://reactjs.org/tutorial/tutorial.html) React

IV. Стиль

1. Архитектура CSS

* [SMACSS](https://smacss.com/)
* [OOCSS](https://github.com/stubbornella/oocss/wiki)
* [Atomic](https://acss.io/)
* [CSS Modules](https://github.com/css-modules/css-modules)
* [BEM](http://getbem.com/)
* [SUITCSS](https://suitcss.github.io/)

1. CSS-фреймворки

* [Semantic UI](https://semantic-ui.com/)
* [Bulma](https://bulma.io/)
* [Materialize, Material UI](https://materializecss.com/)
* [Bootstrap](https://getbootstrap.com/)

1. CSS-препроцессоры

* [Less](http://lesscss.org/)
* [PostCSS](https://postcss.org/)
* [Sass/CSS](https://sass-lang.com/)
* [Stylus](http://stylus-lang.com/)

1. CSS в JS

* [Styled Components](https://www.styled-components.com/)
* [Radium](https://formidable.com/open-source/radium/)
* [Emotion](https://emotion.sh/)
* [JSS](https://cssinjs.org/?v=v10.0.0-alpha.7)
* [Aphrodite](https://github.com/Khan/aphrodite)

V. Познакомиться с популярными инструментами разработки для React Developer

1. Выполнение задач

* [npm-скрипты](https://docs.npmjs.com/misc/scripts)
* [gulp](https://gulpjs.com/)
* [Parcel](https://parceljs.org/)
* [Webpack](https://webpack.js.org/)
* [Rollup](https://rollupjs.org/guide/en)

1. Менеджеры пакетов

* [pnpm](https://pnpm.js.org/)
* [npm](https://www.npmjs.com/)
* [yarn](https://yarnpkg.com/lang/en/)

VI. Управление состоянием

1. [Состояние компонента](https://reactjs.org/docs/faq-state.html) / [Context API](https://reactjs.org/docs/context.html" \t "_blank)
2. [Redux](https://redux.js.org/)
3. Асинхронность
   * [Redux Thunk](https://github.com/reduxjs/redux-thunk)
   * [Redux Better Promise](https://github.com/Lukasz-pluszczewski/redux-better-promise)
   * [Redux Saga](https://redux-saga.js.org/)
   * [Redux Observable](https://redux-observable.js.org/)
4. Библиотеки-помощники
   * [Rematch](https://rematch.gitbooks.io/rematch/#getting-started)
   * [Reselect](https://github.com/reduxjs/reselect)
5. Хранение данных
   * [Redux Persist](https://github.com/rt2zz/redux-persist)
   * [Redux Phoenix](https://github.com/adam-golab/redux-phoenix)
   * [Redux Form](https://redux-form.com/8.1.0/)
6. [MobX](https://mobx.js.org/)

VII. Маршрутизация

* [React-Router](https://reacttraining.com/react-router/)
* [Router5](https://router5.js.org/)
* [Redux-First Router](https://github.com/faceyspacey/redux-first-router)
* [Reach Router](https://reach.tech/router)

VIII. Типизация

* [PropTypes](https://reactjs.org/docs/typechecking-with-proptypes.html)
* [TypeScript](https://www.typescriptlang.org/)
* [Flow](https://flow.org/en/)

IX. Наборы полезных функций

* [Lodash](https://lodash.com/)
* [Moment](https://momentjs.com/)
* [classnames](https://github.com/JedWatson/classnames)
* [Numeral](http://numeraljs.com/)
* [RxJS](http://reactivex.io/)
* [ImmutableJS](https://facebook.github.io/immutable-js/)
* [Ramda](https://ramdajs.com/)

X. Формы

* [Formsy](https://github.com/formsy/formsy-react)
* [Redux Form](https://redux-form.com/8.1.0/)
* [Final Form](https://github.com/final-form/final-form)
* [Formik](https://github.com/jaredpalmer/formik)

XI. Интернационализация

* [React i18next](https://react.i18next.com/)
* [React Intl](https://github.com/yahoo/react-intl)

XII. API-клиенты

* [Fetch](https://developer.mozilla.org/en-US/docs/Web/API/Fetch_API)
* [Relay](https://facebook.github.io/relay/)
* [axios](https://github.com/axios/axios)
* [urql](https://github.com/FormidableLabs/urql)
* [Apollo](https://www.apollographql.com/docs/react/)
* [SuperAgent](https://visionmedia.github.io/superagent/)

XIII. Разработка десктопных приложений

* [React Native Windows](https://github.com/Microsoft/react-native-windows)
* [Proton Native](https://proton-native.js.org/#/)
* [Electron](https://electronjs.org/)

XIV. Разработка мобильных приложений

* [React Native](https://facebook.github.io/react-native/)
* [Cordova](https://cordova.apache.org/) / [Phonegap](https://phonegap.com/" \t "_blank)

XV. Серверная часть

* [Next.js](https://nextjs.org/)
* [After.js](https://github.com/jaredpalmer/after.js)
* [Rogue](https://github.com/alidcastano/rogue.js)

XVI. Тестирование

1. Модульное тестирование
   * [Sinon](https://sinonjs.org/)
   * [Jest](https://jestjs.io/)
   * [Enzyme](https://airbnb.io/enzyme/)
   * [Tape](https://github.com/substack/tape)
   * [Mocha](https://mochajs.org/)
   * [AVA](https://github.com/avajs/ava)
   * [Chai](https://www.chaijs.com/)
2. End-to-end тестирование
   * [Selenium](https://www.seleniumhq.org/), [Webdriver](https://webdriver.io/" \t "_blank)
   * [Puppeteer](https://pptr.dev/)
   * [Cypress](https://www.cypress.io/)
   * [Nightwatch.js](http://nightwatchjs.org/)
   * [Cucumber.js](https://github.com/cucumber/cucumber-js)

XVII. Виртуальная реальность

1. [React 360](https://facebook.github.io/react-360/)

XVIII. Бэкенд-фреймворки

1. [React on Rails](https://shakacode.gitbooks.io/react-on-rails/content/)

XIX. Генераторы

1. [Gatsby для React Developer](https://www.gatsbyjs.org/)

Другие материалы по теме:

* [Как кодить на JavaScript в 2019? Тренды, тенденции, предсказания](https://proglib.io/p/javascript-2019/)
* [Объектная модель документа: что такое DOM и чем не является?](https://proglib.io/p/what-is-dom/)

**Теги**

[Дайджесты](https://proglib.io/p/category/learning/digests/)