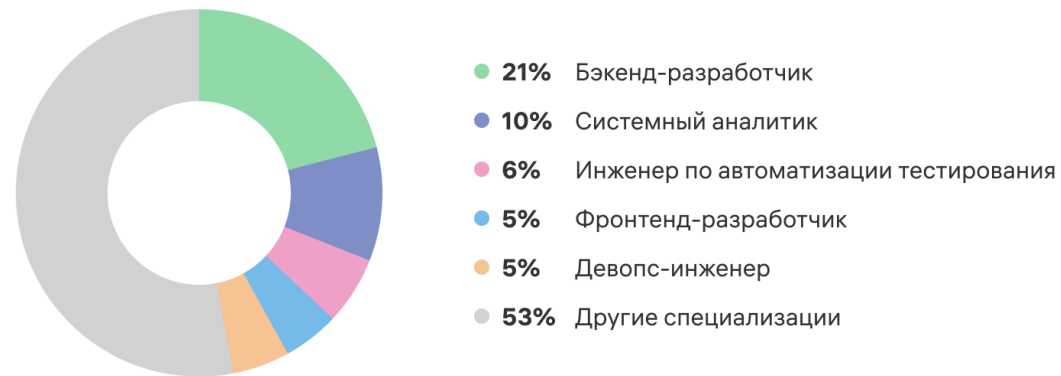


Веб-программирование

Вводное занятие

Веб-разработка в 2023

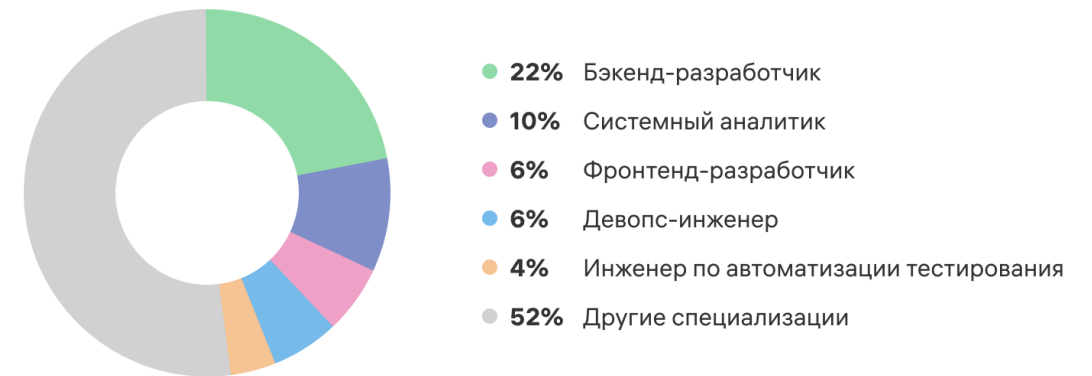
Топ 5 специализаций, которые искали в апреле



Активность найма на IT-рынке в апреле 2023

Хабр Карьера

Топ 5 специализаций, которые искали в мае 2023





Активность найма на IT-рынке в мае 2023

Хабр Карьера

https://habr.com/ru/companies/habr_career/articles/733592/

Веб-разработка в 2023

Sep 2023	Sep 2022	Change	Programming Language		Ratings	Change
1	1			Python	14.16%	-1.58%
2	2			C	11.27%	-2.70%
3	4	▲		C++	10.65%	+0.90%
4	3	▼		Java	9.49%	-2.23%
5	5			C#	7.31%	+2.42%
6	7	▲		JavaScript	3.30%	+0.48%
7	6	▼		Visual Basic	2.22%	-2.18%
8	10	▲		PHP	1.55%	-0.13%
9	8	▼		Assembly language	1.53%	-0.96%
10	9	▼		SQL	1.44%	-0.57%

<https://www.tiobe.com/tiobe-index/>

Веб-разработка в 2023



https://madnight.github.io/github/#/pull_requests/2023/2

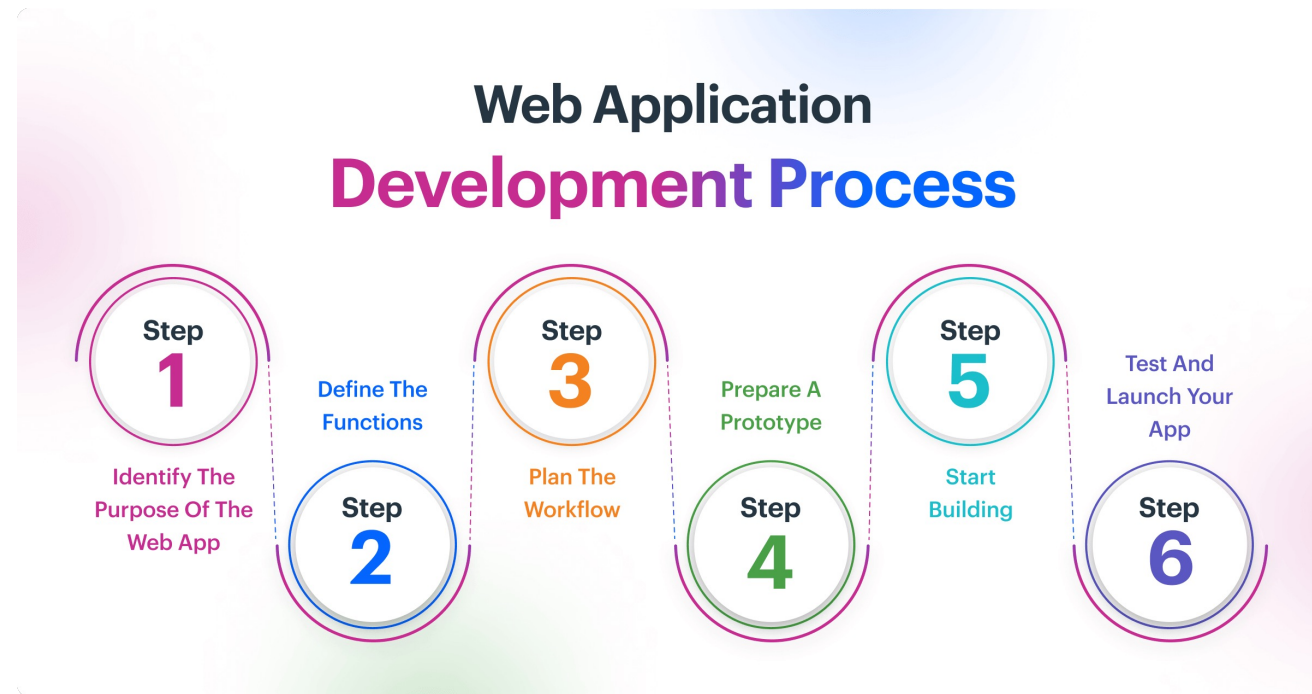
Веб-разработка в 2023



<https://madnight.github.io/githut/#/stars/2023/2>

Зачем эти знания?

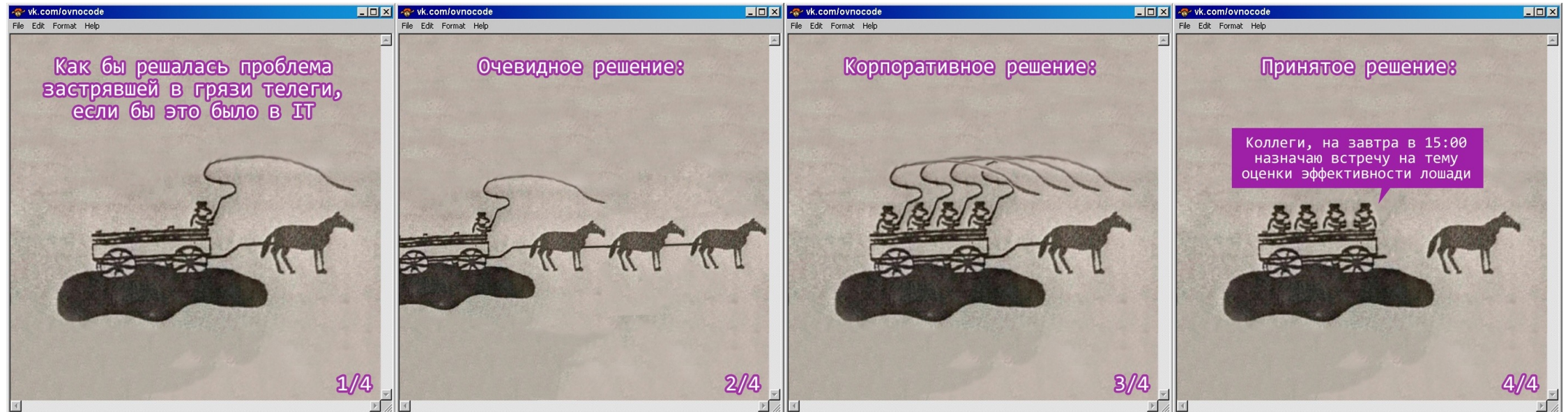
- Понимание того, что происходит на каждом этапе разработки веб-приложений



<https://kissflow.com/application-development/web-application-development/>

Зачем эти знания?

- Более эффективное планирование фичей



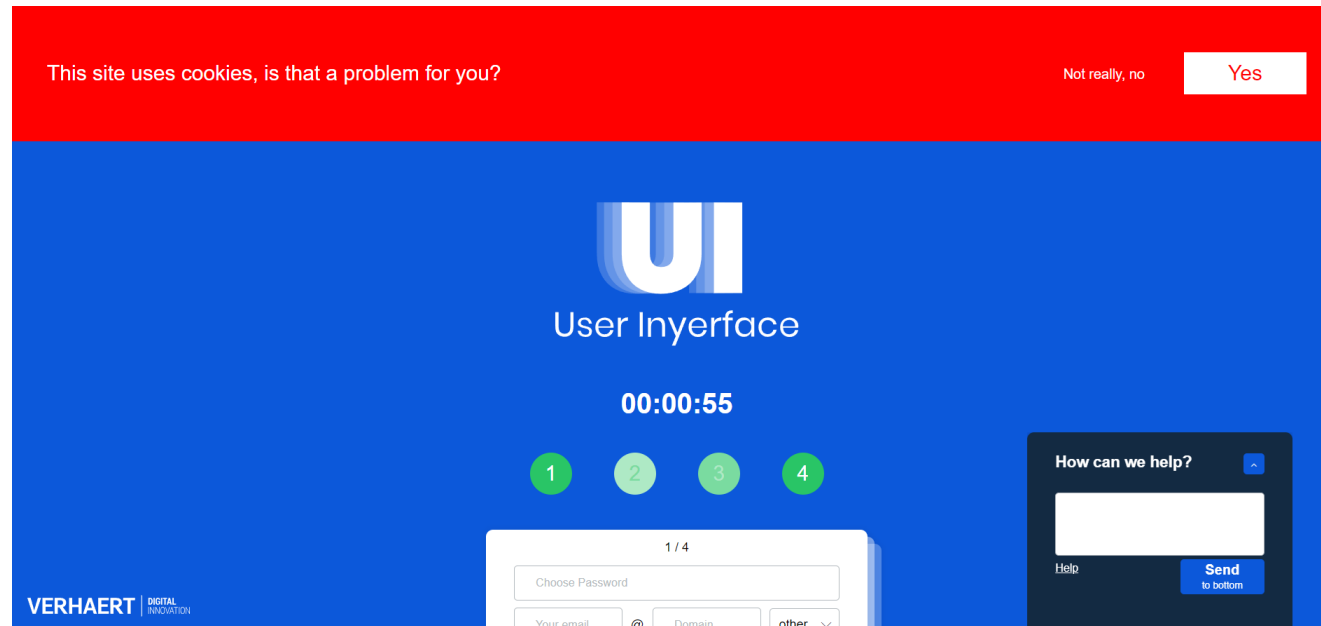
Зачем эти знания?

- Меньше шансов взаимодействия с недобросовестными разработчиками



Зачем эти знания?

- Более грамотное проектирование дизайна



<https://userinyerface.com/game.html>

Зачем эти знания?

- Веб-разработка — это интересно

The screenshot shows a web browser window with a basic HTML page. The page content is as follows:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>
    Basic Example to Describe JavaScript
  </title>
</head>
<body>
  <!-- JavaScript code can be embedded inside
  head section or body section -->
  <script>
    console.log("Welcome to GeeksforGeeks");
  </script>
</body>
</html>
```

Below the code, the output of the JavaScript code is displayed in the console:

```
Welcome to GeeksforGeeks
```

On the right side of the browser, the DevTools console is open, showing several error messages:

- invalid tag id
- DevTools failed to load source map: Could not load content for https://as
- sets.a-mo.net/js/cframe.js.map: HTTP error: status code 403,
- net::ERR_HTTP_RESPONSE_CODE_FAILURE
- GET https://id.rlcdn.com/711916.gif?ct=4&cv= 451 711916.gif:1
- Failed to load resource: the server responded with a status of 451 ()
- Failed to load resource: the server responded with a status of 451 ()
- DevTools failed to load source map: Could not load content for https://as
- sets.a-mo.net/js/cframe.js.map: HTTP error: status code 403,
- net::ERR_HTTP_RESPONSE_CODE_FAILURE
- Mixed Content: The page at 'https://onetag-sys.com/usyn onetag-sys.com:1
- c/?pubId=5adb88524e24e50&us_privacy=1---' was loaded over HTTPS, but
- requested an insecure element 'http://sync.adkernel.com/user-sync?zone=17
- 5005&n=UEWia0ktWahj_csosGhPwFzQg5L096L-JN4002d_Y5A'. This request was
- automatically upgraded to HTTPS, For more information see https://blog.ch
- romium.org/2019/10/no-more-mixed-messages-about-https.html
- Mixed Content: The page at 'https://onetag-sys.com/usyn onetag-sys.com:1
- c/?pubId=5adb88524e24e50&us_privacy=1---' was loaded over HTTPS, but
- requested an insecure element 'http://sync.adkernel.com/user-sync?zone=17
- 5005&n=UEWia0ktWahj_csosGhPwFzQg5L096L-JN4002d_Y5A'. This request was
- automatically upgraded to HTTPS, For more information see https://blog.ch
- romium.org/2019/10/no-more-mixed-messages-about-https.html
- GET https://id.rlcdn.com/711916.gif?ct=4&cv= 451 711916.gif:1
- Failed to load resource: the server responded with a status of 451 ()
- Failed to load resource: the server responded with a status of 451 ()
- GET https://sync.adkernel.com/user-sync?zone=175005&n=U user-sync:1
- EWia0ktWahj_csosGhPwFzQg5L096L-JN4002d_Y5A 400 (Bad Request)
- Failed to load resource: the server responded with a status of 400 (Bad Request)
- GET https://ids-sync.com/i/495/0.gif?callback=https%3A% 152mediaweb:1
- 2F%2Fsync.quantumdex.io%2Fsetuid%3Fbldder%3Dinmobi%26guid%3D%7B7D5D10%7D
- 410

<https://www.geeksforgeeks.org/introduction-to-javascript/?ref=roadmap>

Структура курса

Неделя 1

- Вводное занятие: рассказ про веб-разработку в целом
- Основные концепции HTML и CSS

Неделя 2

- Ответы на вопросы по теме HTML и CSS
- Работа с фигмой в контексте верстки
- ДЗ: сверстать своё приложение (1 одна неделя)

Структура курса

Неделя 3

- Ответы на вопросы по выполненному заданию
- Основные концепции JavaScript (+ домашнее обучение)
- ДЗ: 10 задачек (2 недели)

Неделя 4

- Углубленный JavaScript

Неделя 5

- Ответы на вопросы по JavaScript
- API и его подключение к проекту
- ДЗ: подключение API к проекту (1 неделя)

Структура курса

Неделя 6

- Ответы на вопросы по API
- Разворачивание проекта в облаке и хостинги
- ДЗ: развернуть проект в облаке (1 неделя)

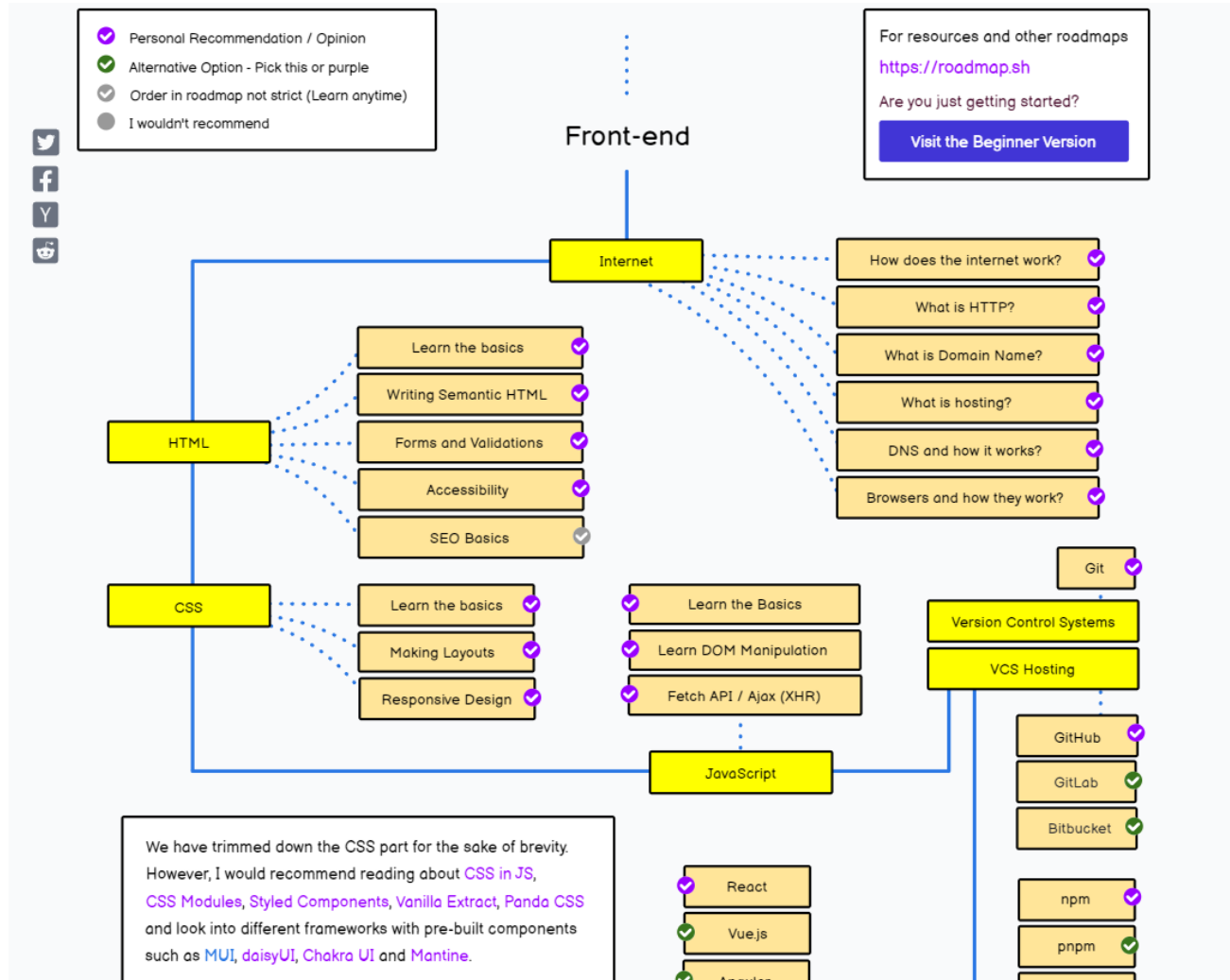
Неделя 7

- Ответы на вопросы про развёртку и хостинги
- Введение в React
- ДЗ: переписать проект на React (на все оставшееся время)

Неделя 8 (и далее)

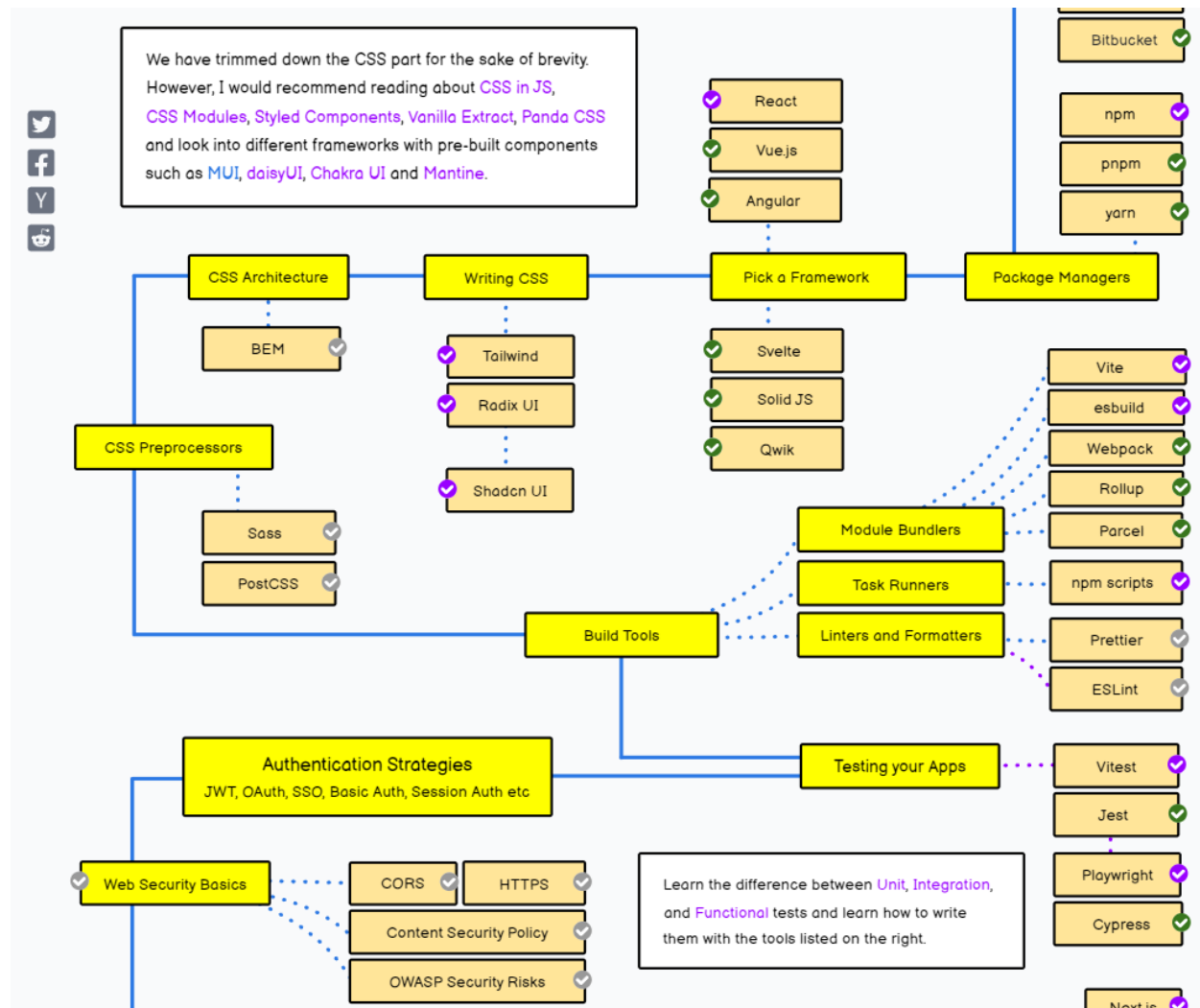
- Введение в TypeScript
- Ответы на вопросы по React и другие

Современная веб-разработка



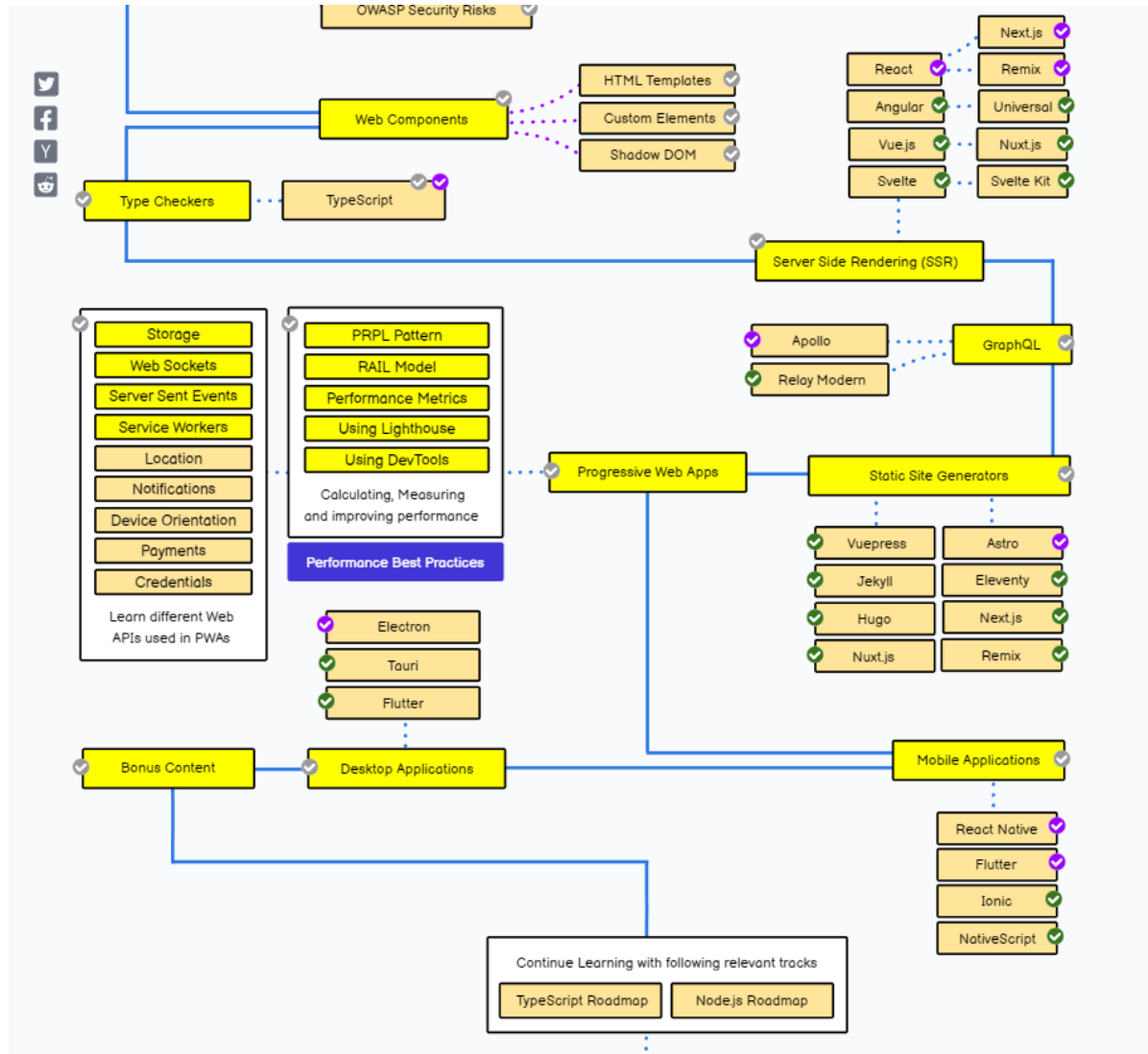
<https://roadmap.sh/frontend>

Современная веб-разработка



<https://roadmap.sh/frontend>

Современная веб-разработка



<https://roadmap.sh/frontend>

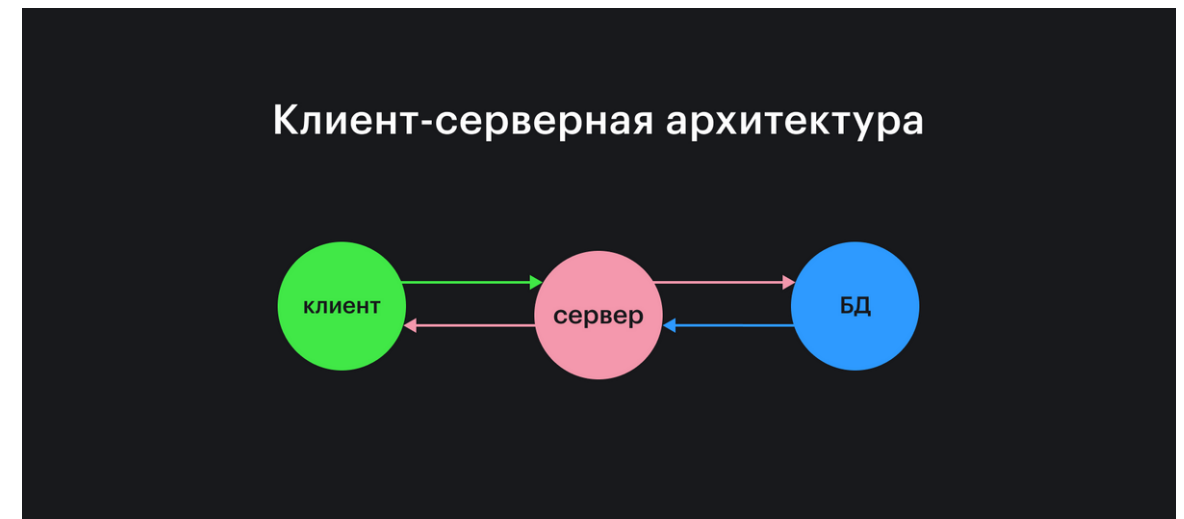
Как работают веб-приложения?

Фронтенд (клиентская часть)

- Берёт данные с бэкенда
- Отрисовывает данные в нужном виде

Бэкенд (серверная часть)

- Хранит данные
- Обработывает данные
- Передаёт данные на клиенты



<https://doka.guide/tools/web-app-works/>

HTML, CSS и JS

- **HTML** — язык разметки для создания структуры веб-страницы и представления контента. Благодаря разметке браузер знает в каком порядке отображать элементы, и что они значат.
- **CSS** — язык каскадных стилей, который задаёт визуальное оформление для HTML.
- **JavaScript** — язык программирования широкого спектра. На нём можно как создавать динамические интерфейсы, так и работать с базами данных и операционными системами.

(взято из <https://doka.guide>)



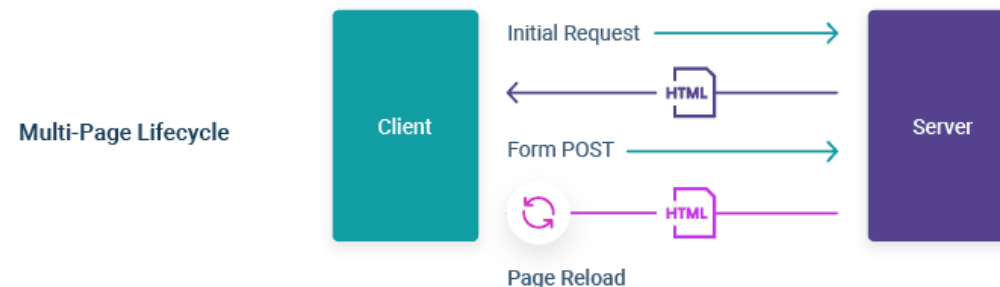
Виды веб-приложений

Статические многостраничные приложения (MPA)

- Набор готовых свёрстанных страниц
- Динамическая генерация HTML на сервере

Одностраничные приложения (SPA)

- Client Side Rendering (CSR)
- Server Side Rendering (SSR)



<https://dev.to/adnanbabakan/deploy-a-single-page-application-with-200-html-2p0f>

Серверный JavaScript

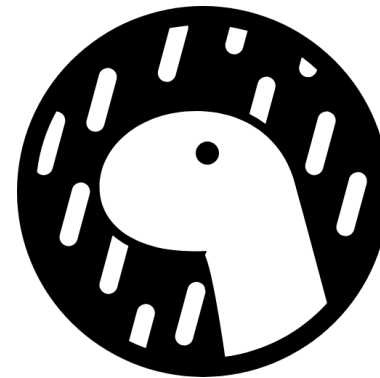
Node.js — это среда выполнения кода JavaScript вне браузера, которая позволяет писать серверный код для веб-страниц и веб-приложений, а также для программ командной строки

Аналоги:

- Deno
- Bun



Node.js



Deno



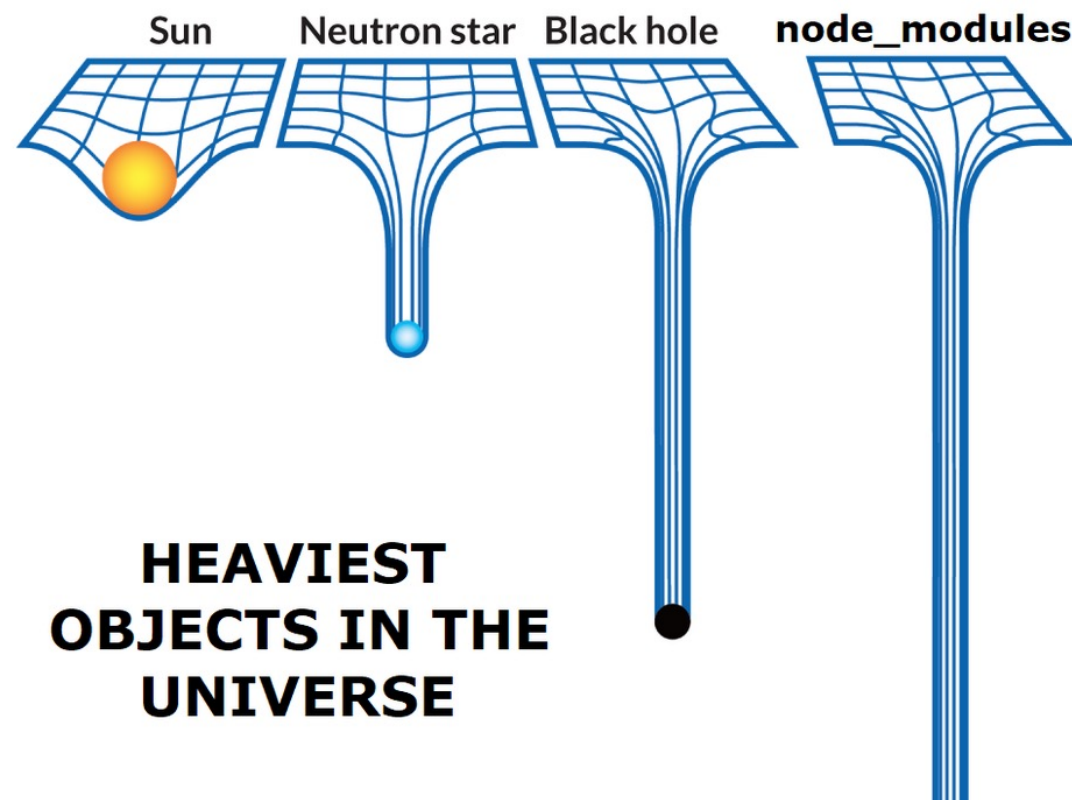
Bun

Пакетные менеджеры

Для удобной работы с ними существуют пакетные менеджеры (менеджеры зависимостей). Они помогают выбрать версию библиотеки и гарантируют безопасность.

Популярные менеджеры:

- npm – стандартный (идёт сразу с Node.js)
- yarn
- pnpm



UI-библиотеки и фреймворки

В мире фронтенда много различных фреймворков для разработки одностраничных приложений.

Большинство из них различаются подходами к разработке и инструментами, которыми они обладают.

Но в основе всех этих фреймворков часто лежат одни и те же концепции.

Основные концепции:

- Реактивность
- Virtual DOM



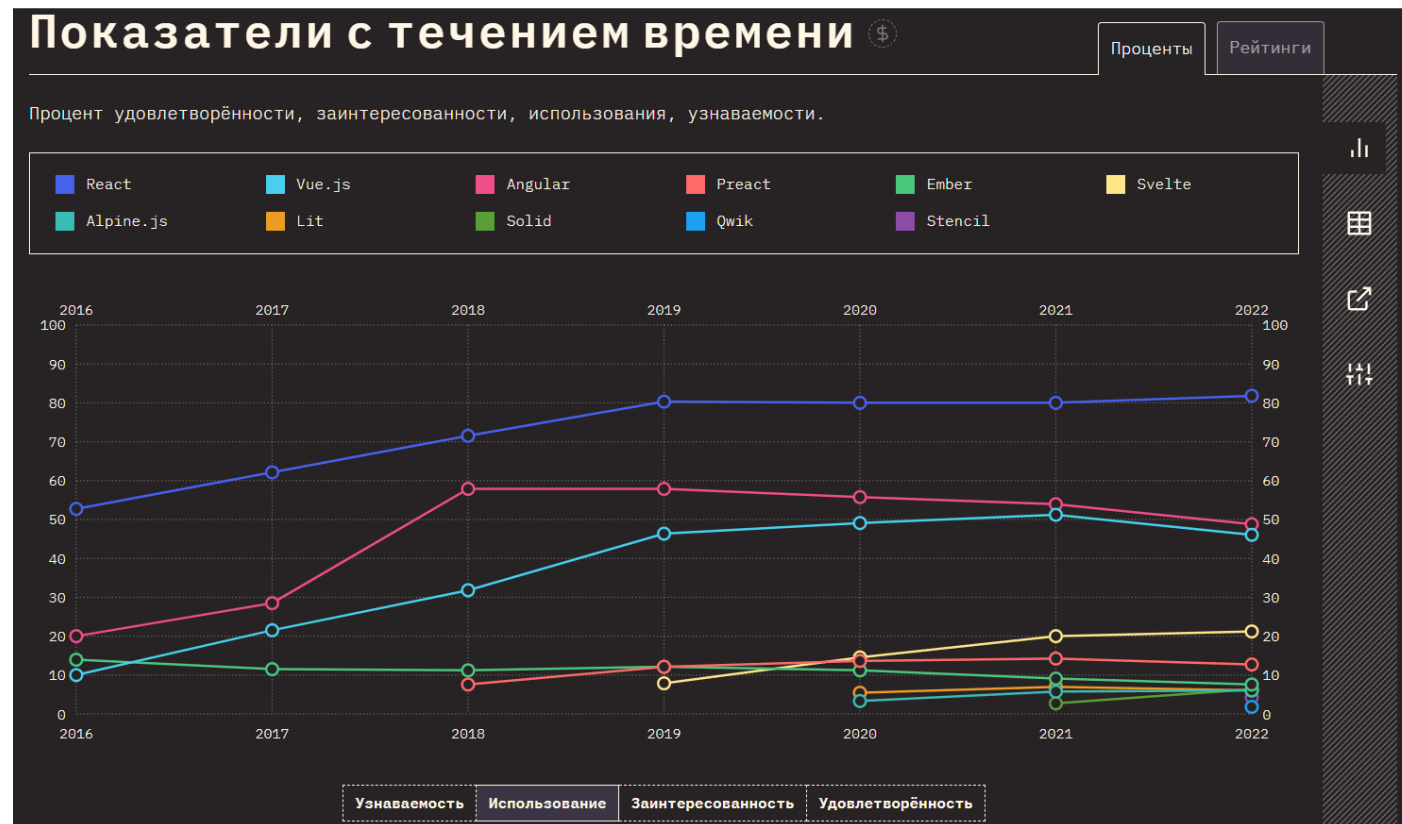
JavaScript Developers



UI-библиотеки и фреймворки

Популярные инструменты:

- React
- Vue
- Angular
- Svelte



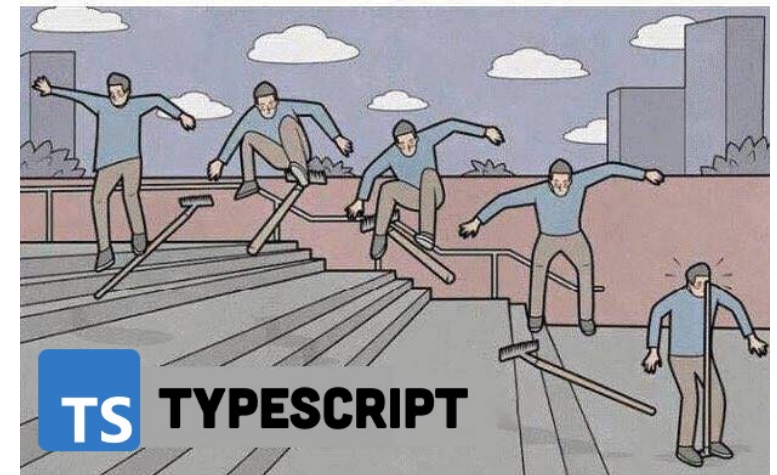
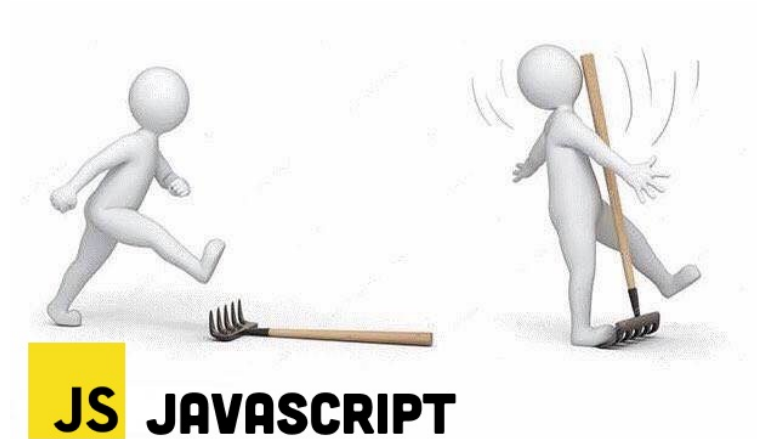
<https://2022.stateofjs.com/ru-RU/libraries/front-end-frameworks/>

TypeScript

В JavaScript слабая динамическая типизация

- При операции с переменными разных типов они будут автоматически приведены к одному
- Любая переменная может произвольно менять свой тип во время выполнения программы

TypeScript = JavaScript + строгая типизация



(Хотя скорее наоборот)

Системы контроля версий (VCS)

VCS (version control systems) — были придуманы, чтобы контролировать изменения исходного кода и комфортно работать над ним большой командой.

Примеры VCS:

- Git:
 - GitHub
 - GitLab
 - BitBucket
- SVN
- Mercurial
- Другие решения



ЛИНТИНГ

Процесс исследования исходного кода без запуска называют статическим анализом или линтингом, а программу, которая это делает — статическим анализатором или линтером.

Самый популярный линтер для JavaScript — **ESLint**

Самый популярный форматтер для JavaScript — **prettier**

Плюсы линтинга:

- Уменьшается разноречивость в коде
- Уменьшается количество ошибок по неосторожности
- Ревью проходит быстрее
- Код во всей кодовой базе становится одинаковым



Сборка

Бандлер (сборщик) — программа, которая упаковывает сложный проект со многими файлами и внешними зависимостями в один (иногда несколько) файл, который будет отправлен браузеру

Популярные сборщики:

- esbuild
- webpack
- parcel

Популярные бойлерплейты:

- create-react-app (cra)
- vite



Тестирование

Тесты делают код более прочным и живучим. Одновременно с этим тесты — это отличная документация, которая не врёт и не устаревает. Также тесты можно использовать как инструмент разработки программы.

Виды тестов:

- Юнит-тесты
- Интеграционные тесты
- e2e-тесты
- Скриншотные тесты



Тестирование

Популярные инструменты для тестирования:

- Jest (юнит, интеграционные и снапшоты)
- MochaJS (юнит)
- Karma (юнит)
- Cypress (интеграционные и e2e)
- Hermione (скриншотные и e2e)
- Playwright (скриншотные и e2e)



No-Code (Low-Code) решения

No-Code-инструменты позволяют обычным пользователям создавать веб-приложения без необходимости написания программного кода (или с минимальной необходимостью – **Low-Code**)

Некоторые no-code платформы:

- Airtable
- Appy Pie
- WordPress
- Notion

Минусы:

- Быстрая разработка в ущерб гибкости
- Ограниченность рамками шаблонов

No Code

No code is the best way to write secure and reliable applications. Write nothing; deploy nowhere.

Getting Started

Start by not writing any code.

This is just an example application, but imagine it doing anything you want. Adding new features is easy too:

The possibilities are endless.

Полезные материалы

<https://htmlbook.ru/> – один из самых толковых ресурсов по HTML и CSS

<https://doka.guide/> – про веб-разработку понятным языком от профессиональных разработчиков

<https://learn.javascript.ru/> – самый подробный учебник по JS с примерами и задачами

<https://developer.mozilla.org/ru/> – документация по HTML, CSS, JS и WebAPI от Mozilla

<https://metanit.com/web/> – учебник с большим количеством примеров прикладных задач