

Знакомство с React

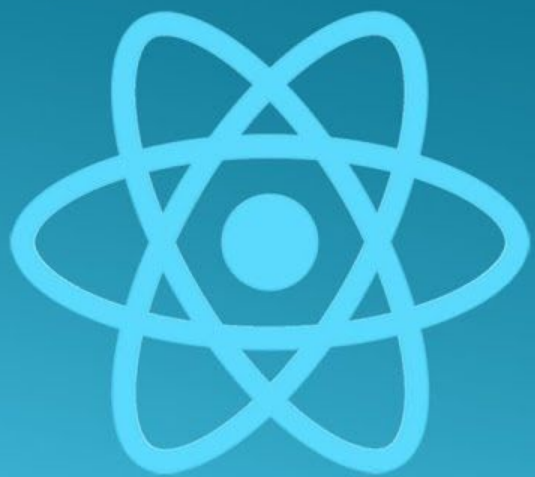
Сбербанк. Фронтенд школа.

Современные фреймворки и Single Page Application



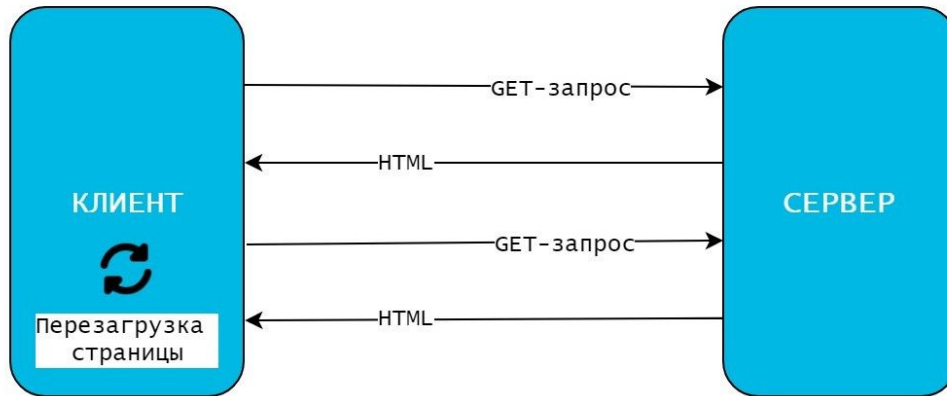
Современные фреймворки и Single Page Application



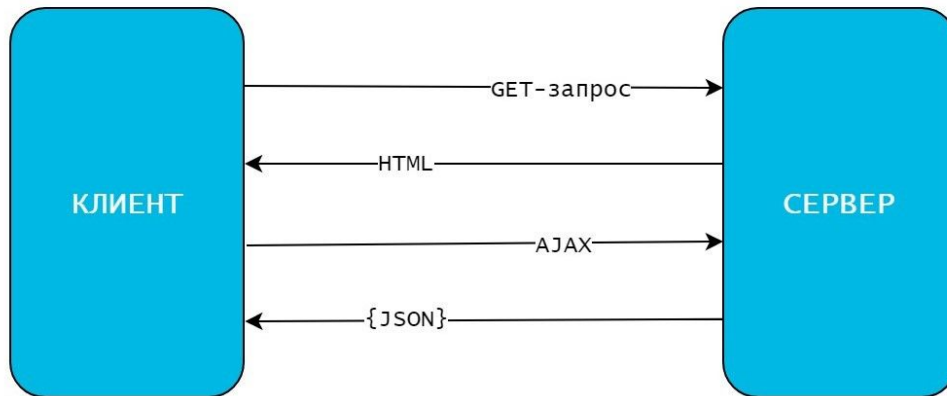


Одностраничное приложение (англ. single page application, SPA) — это веб-приложение или веб-сайт, использующий единственный HTML-документ как оболочку для всех веб-страниц и организующий взаимодействие с пользователем через динамически подгружаемые *HTML*, *CSS*, *JavaScript*, обычно посредством *AJAX*.

Традиционные сайты

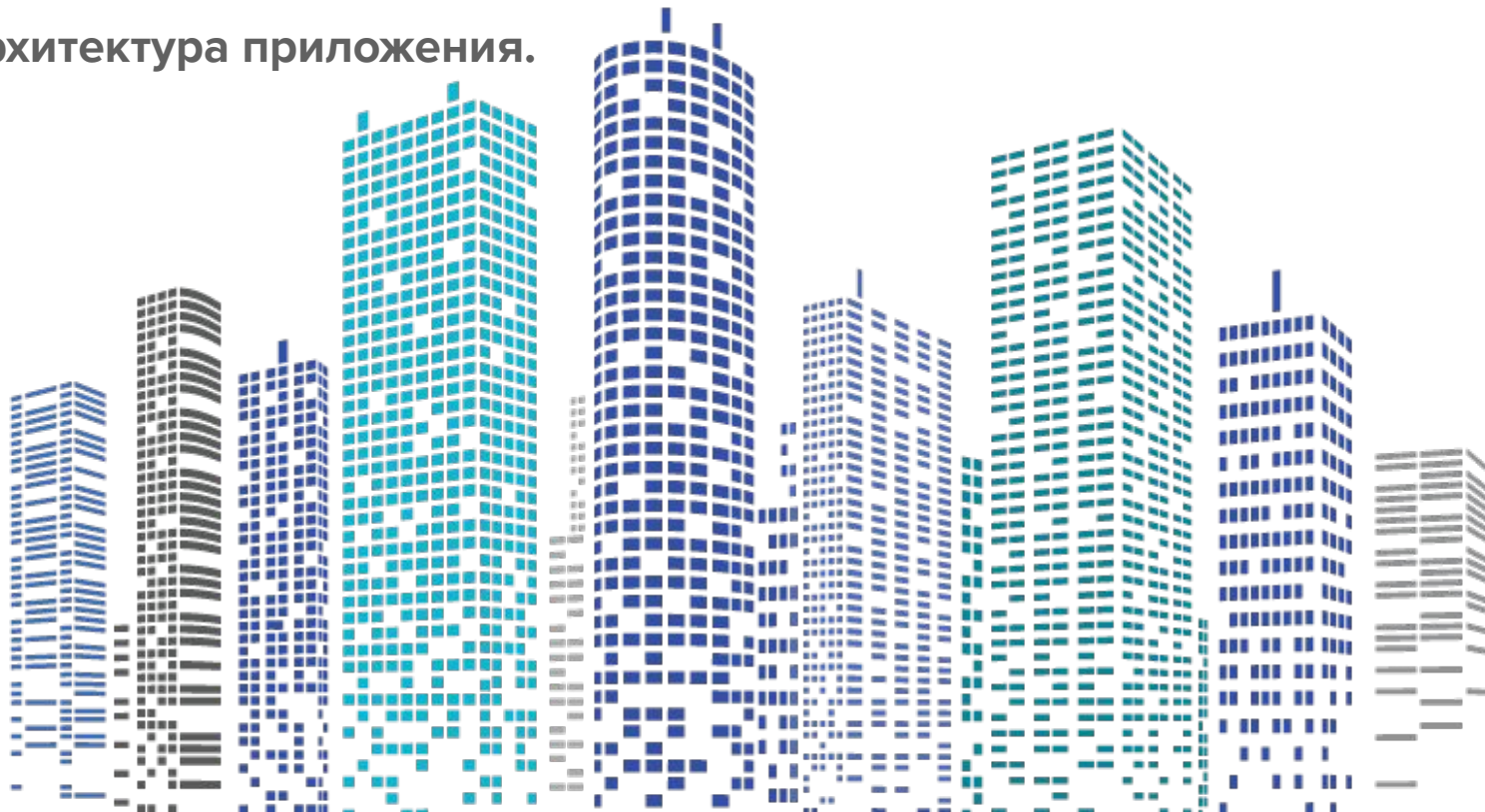


SPA-сайты



Современные фреймворки и Single Page Application

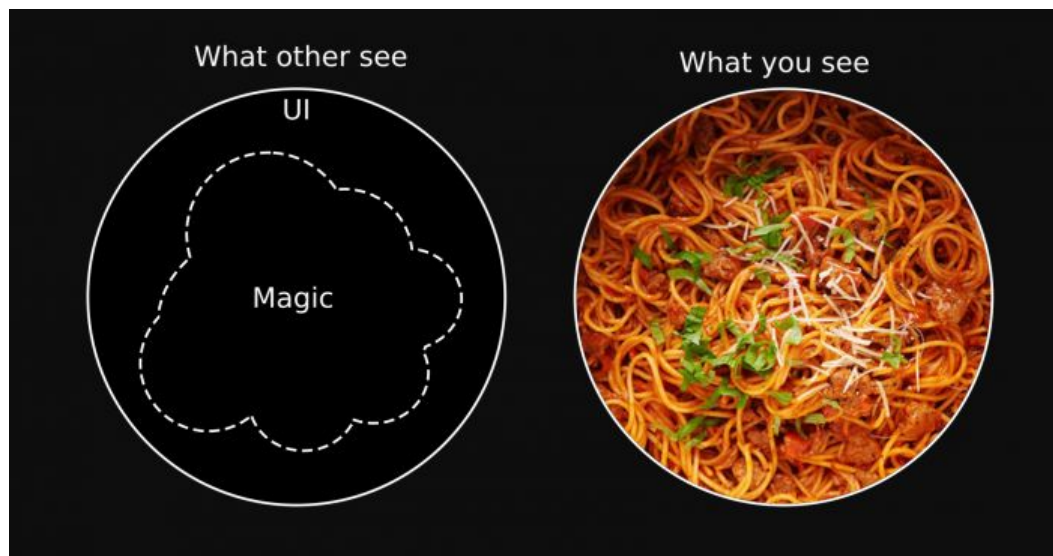
Архитектура приложения.



Современные фреймворки и Single Page Application

Архитектура приложения.

Старые-добрые времена, вдохновленные jQuery



Современные фреймворки и Single Page Application

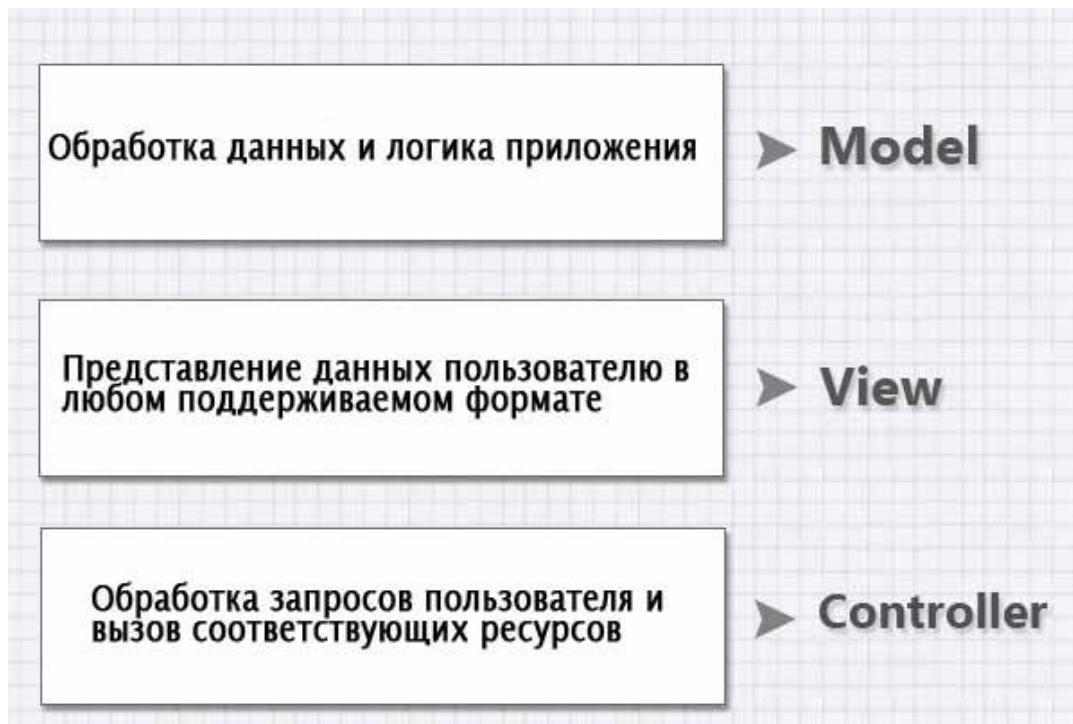
Архитектура приложения.

Современность...

Попытки разделения представлений от логики.

(MVC, MVP, MVW, MVVM... etc)

Современные фреймворки и Single Page Application



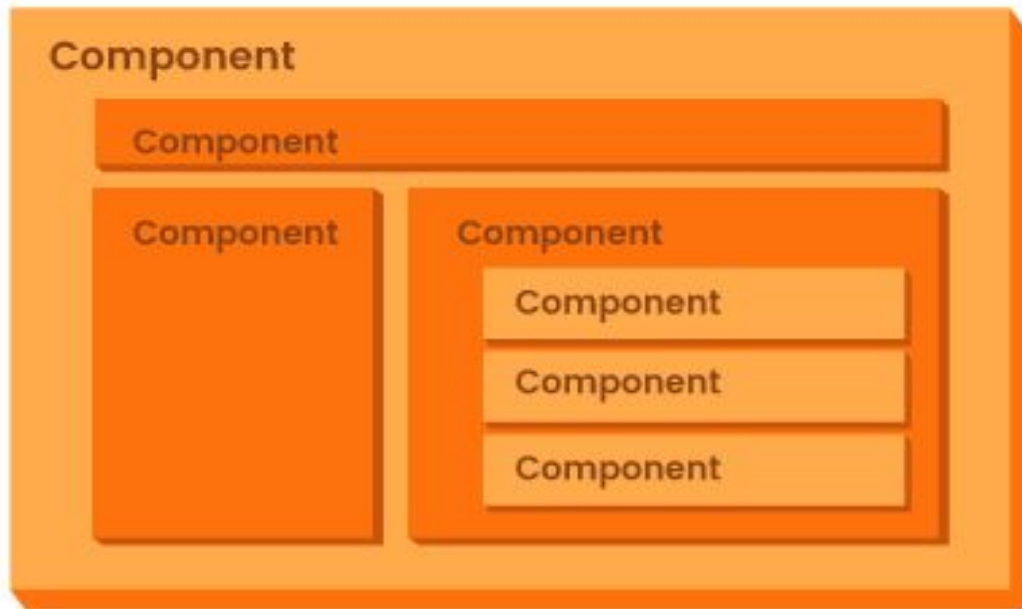
Современные фреймворки и Single Page Application

Компонентный подход:

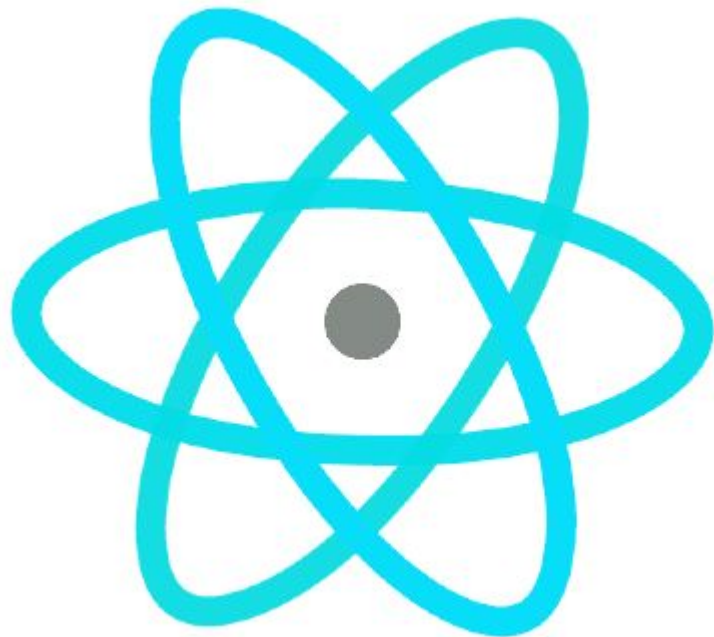
**смысл в разделении кода приложения на
независимые, слабосвязанные и переиспользуемые
компоненты.**

Слабые связи и декомпозиция.

Современные фреймворки и Single Page Application



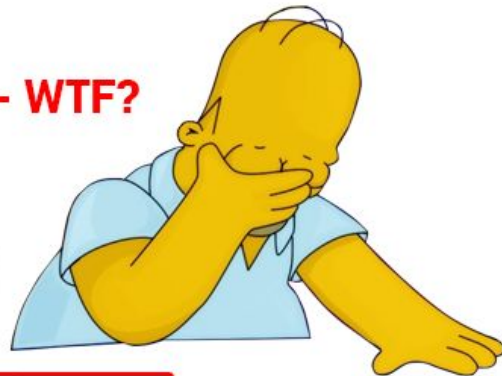
Современные фреймворки и Single Page Application



Современные фреймворки и Single Page Application

```
1  import React from 'react';
2
3  import './user.scss';
4
5  class User extends React.Component {
6
7    render() {
8      return <div className='user'>
9        <img src={this.props.avatar} />
10       <span>{this.props.name}</span>
11      </div>;
12    }
13  }
14
15  export default User;
```

- WTF?



- Уродливо

- А как же разделение
ответственности?!

Современные фреймворки и Single Page Application

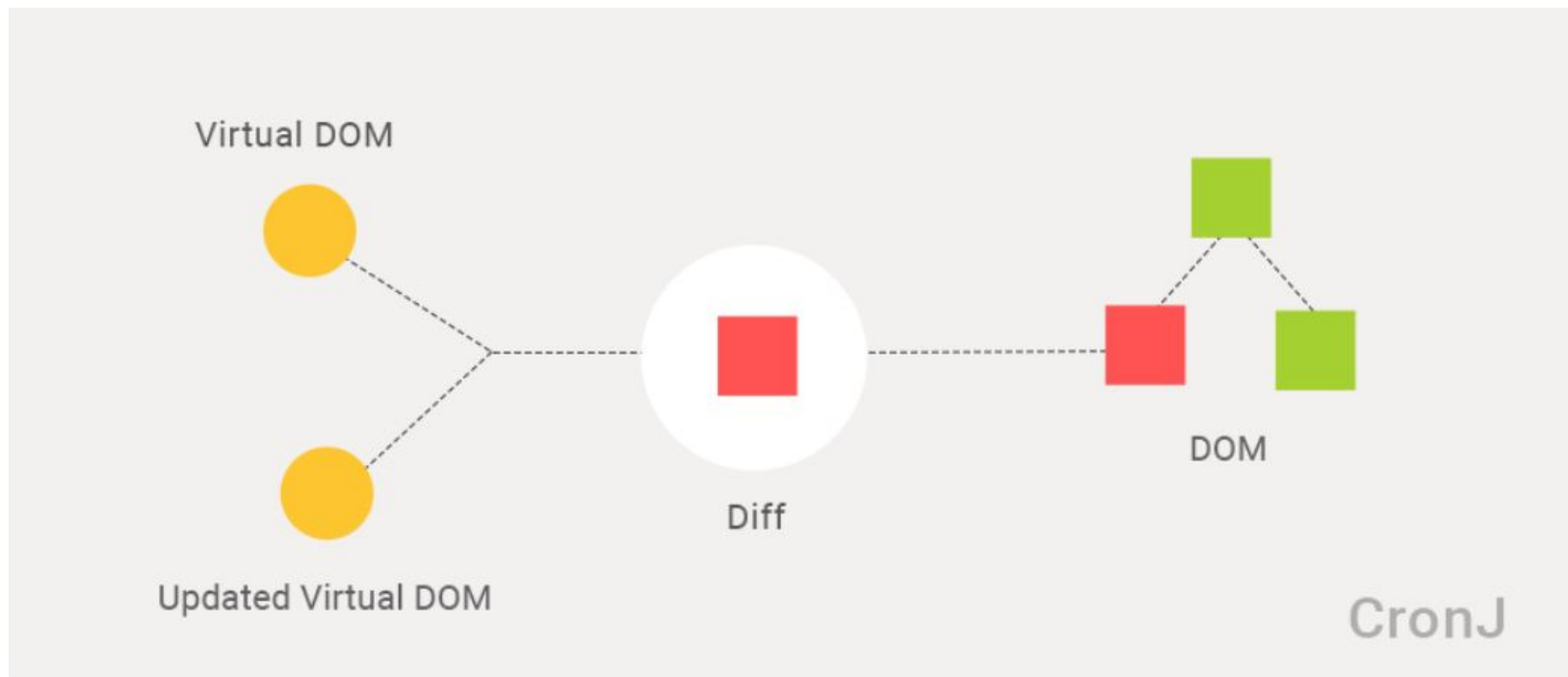
Преимущества React

- 1) Быстрый
- 2) Компонентный
- 3) Расширяемый
- 4) Гибкий
- 5) Популярный



Современные фреймворки и Single Page Application

Virtual DOM



Современные фреймворки и Single Page Application

Virtual DOM

- Вместо того, чтобы взаимодействовать с DOM напрямую, мы работаем с его легковесной копией
- Мы можем вносить изменения в копию, исходя из наших потребностей, а после этого применять изменения к реальному DOM
- При этом происходит сравнение DOM-дерева с его виртуальной копией, определяется разница и запускается перерисовка того, что было изменено

Современные фреймворки и Single Page Application

JSX

```
const element = <h1>Привет, мир!</h1>;
```

<https://jsbin.com/zuqijifufo/edit?html,js,output>

Современные фреймворки и Single Page Application

REACT CREATE APP

```
npx create-react-app my-app  
cd my-app  
npm start
```

<https://github.com/facebook/create-react-app>

Современные фреймворки и Single Page Application

PROPS / STATE

PROPS - неизменяемые свойства, получаемые извне.

STATE - состояние компонента, которое он может изменять.

<https://jsbin.com/cawezofodu/edit?html,js,output>

Современные фреймворки и Single Page Application

PROPS / STATE

- 1) Кофейные зерна
- 2) Молоко
- 3) Какой напиток будет приготовлен
- 4) Крепость кофе
- 5) Температура приготовления
- 6) Кружка



Современные фреймворки и Single Page Application

Обработчики событий

- События в React именуются в стиле camelCase вместо нижнего регистра.
- С JSX вы передаёте функцию как обработчик события вместо строки.

Например, в HTML:

```
<button onclick="activateLasers()">  
  Активировать лазеры  
</button>
```

В React немного иначе:

```
<button onClick={activateLasers}>  
  Активировать лазеры  
</button>
```

Современные фреймворки и Single Page Application

Обработчики событий

- 1) Привязка в конструкторе (OLDSCHOOOOOOOOL)
(<https://codepen.io/gaearon/pen/xEmzGg?editors=0010>)
- 2) Синтаксис полей классов (Модно-современно)
(<https://jsbin.com/lofayidavi/edit?html,js,output>)
- 3) Стрелочные функции в колбэке (тоже модно, но могут быть проблемы)
(<https://jsbin.com/lofayidavi/edit?html,js,output>)

Современные фреймворки и Single Page Application

Практика

<https://jsbin.com/yiqewepivo/edit?css.js>

Современные фреймворки и Single Page Application

Списки и ключи

Списки - и в Африке списки.

Ключи - помогают React определять, какие элементы были изменены, добавлены или удалены. Их необходимо указывать, чтобы React мог сопоставлять элементы массива с течением времени

<https://codepen.io/gaearon/pen/jrXYRR?editors=0011>

Современные фреймворки и Single Page Application

Списки и ключи



<https://codepen.io/gaearon/pen/jrXYRR?editors=0011>

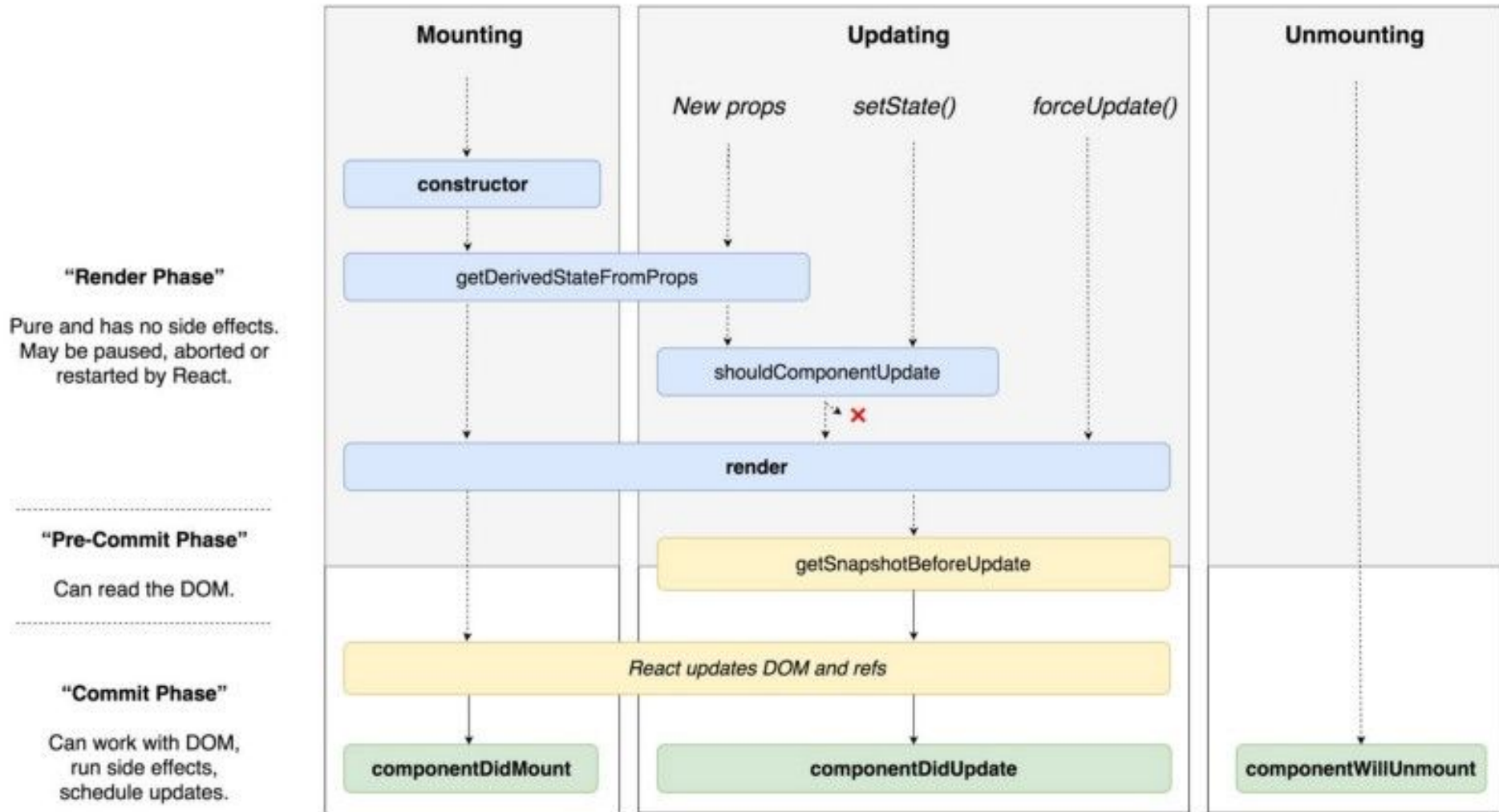
Современные фреймворки и Single Page Application

Списки и ключи



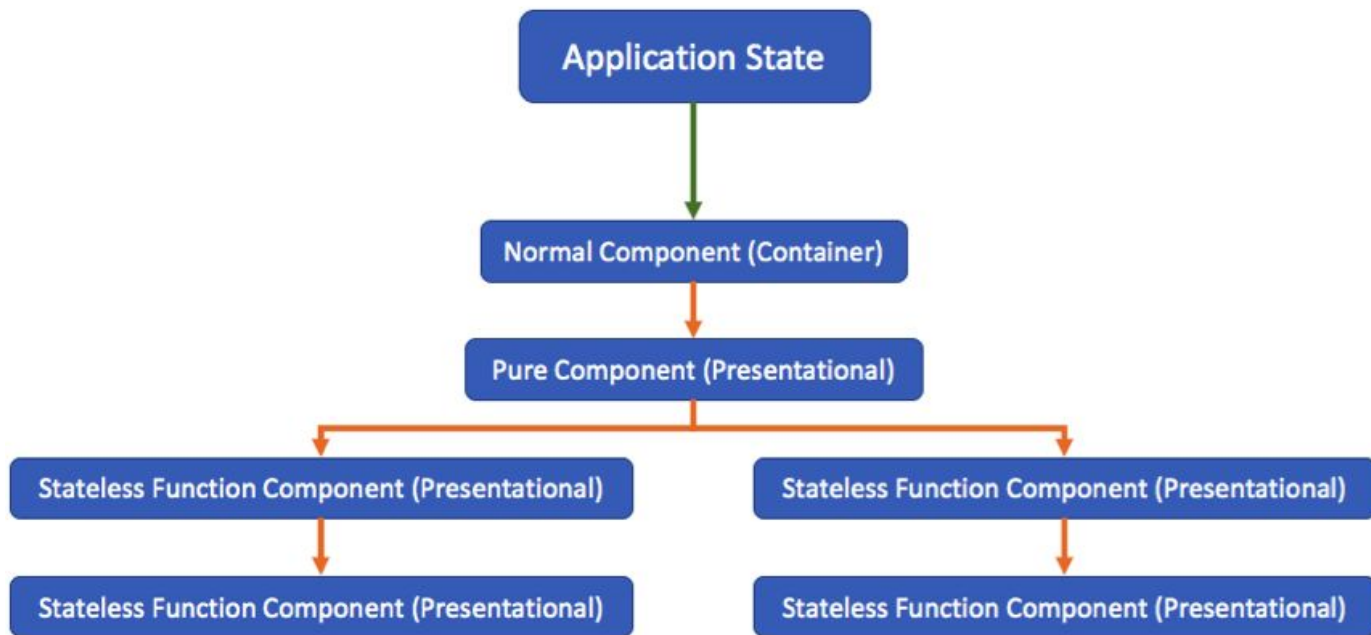
<https://codepen.io/gaearon/pen/jrXYRR?editors=0011>

Жизненный цикл компонента



Современные фреймворки и Single Page Application

Компоненты. Функциональные, Pure, Обычные...



Внеклассное чтение

- 1) Шпаргалка по React (<https://devhints.io/react>)
- 2)
- 3) Жизненный цикл реакт - компонента ([документация](#), [подробнее](#))
- 4) 6 полезных подсказок (<https://quasi-art.ru/library/it/6-tips-for-react-beginners>)
- 5) Ключи, списки (<https://habr.com/ru/company/hh/blog/352150/>)
- 6) Подробнее о компонентах
(<https://code.tutsplus.com/ru/tutorials/stateful-vs-stateless-functional-components-in-react--cms-29541>)