

< Компоненты />

# Всем привет!



## Крайнов Алексей

Frontend developer Менеджер Дизайнер

Не миллиардер Не филантроп



[@design\\_system\\_ru](https://t.me/design_system_ru)



[@krainovlife](https://t.me/krainovlife)



[prog.msk.ru](https://prog.msk.ru)



Магазин



Каталог



Корзина

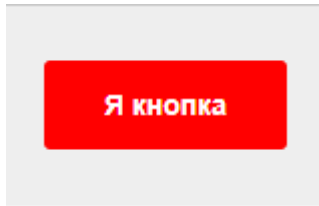
Я кнопка

**Тут контент !!!!!!!**

# Header

```
<div class="header">
  <div class="header__wrap">
    <div class="header__logo">
      
    </div>
    <div class="header__panel">
      <div class="header__panel-item">
        <i class="header__panel-icon"></i>
        <div class="header__panel-text">Магазин</div>
      </div>
      <div class="header__panel-item">
        <i class="header__panel-icon"></i>
        <div class="header__panel-text">Каталог</div>
      </div>
      <div class="header__panel-item">
        <i class="header__panel-icon"></i>
        <div class="header__panel-text">Корзина</div>
      </div>
      <a href="#" class="header__button">Я кнопка</a>
    </div>
  </div>
</div>
```

# Header



```
document.querySelector('header__button').addEventListener("click", function () {  
  document.querySelector(this).classList.add("header__button-active");  
});
```

# Что то не так?

- Плохая масштабируемость
- Дублирование кода
- Отсутствие единого стиля
- Сложная поддержка кода
- Не модно

# Button

```
<a href="#" class="button ">Независимая кнопка</a>
```

Независимая кнопка

```
<a href="#" class="button button_theme_green">Независимая кнопка</a>
```

Независимая кнопка

```
<a href="#" class="button button_theme_green button_size_default">Независимая кнопка</a>
```

Независимая кнопка

```
document.querySelector('.button').addEventListener("click", function () {  
  document.querySelector(this).classList.add(".button_theme_red");  
});
```

Независимая кнопка

# Header

```
<div class="header">
  <div class="header__wrap">
    <div class="header__logo">
      
    </div>
    <div class="header__panel">
      <div class="header__panel-item">
        <i class="header__panel-icon"></i>
        <div class="header__panel-text">Магазин</div>
      </div>
      <div class="header__panel-item">
        <i class="header__panel-icon"></i>
        <div class="header__panel-text">Каталог</div>
      </div>
      <div class="header__panel-item">
        <i class="header__panel-icon"></i>
        <div class="header__panel-text">Корзина</div>
      </div>
      <a href="#" class="button button_theme_green button_size_default">Я кнопка</a>
    </div>
  </div>
</div>
```



# Icon

```
<i class="icon icon_name_blabla icon_size_default"></i>
```



```
<a href="#" class="button button_theme_green button_size_default button_icon">  
  <i class="icon icon_name_blabla icon_size_default"></i>  
  <span class="button__text">Независимая кнопка</span>  
</a>
```



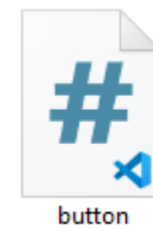
Независимая кнопка

# Icon

```
<div class="header">
  <div class="header__wrap">
    <div class="header__logo">
      
    </div>
    <div class="header__panel">
      <div class="header__panel-item">
        <i class="header__panel-icon icon icon_name_blabla icon_size_default"></i>
        <div class="header__panel-text">Магазин</div>
      </div>
      <div class="header__panel-item">
        <i class="header__panel-icon icon icon_name_blabla icon_size_default"></i>
        <div class="header__panel-text">Каталог</div>
      </div>
      <div class="header__panel-item">
        <i class="header__panel-icon icon icon_name_blabla icon_size_default"></i>
        <div class="header__panel-text">Корзина</div>
      </div>
      <a href="#" class="button button_theme_green button_size_default">Я кнопка</a>
    </div>
  </div>
</div>
```

# Что нужно

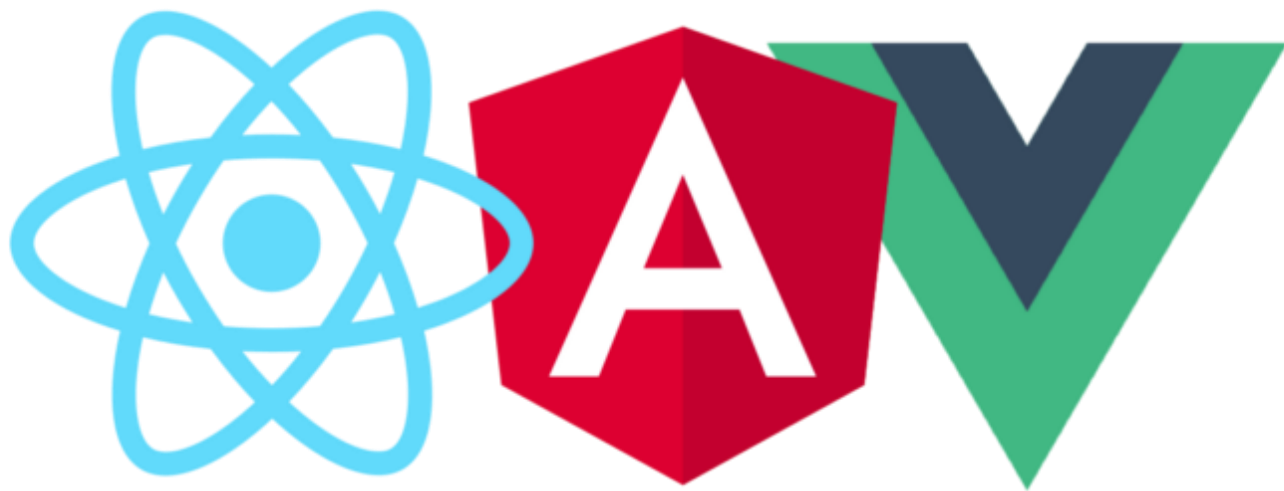
- HTML
- CSS
- JS



# Компоненты

- Полностью изолированные код
- Единая стилизация нативных элементов управления
- Удобная и простая масштабируемость
- Гибкое использование в любых проектах

# Возможности



# Angular

- Angular используется вместе с Typescript.
- Односторонняя привязка данных
- Структура и архитектура
- Подробная документация

# Vue.js

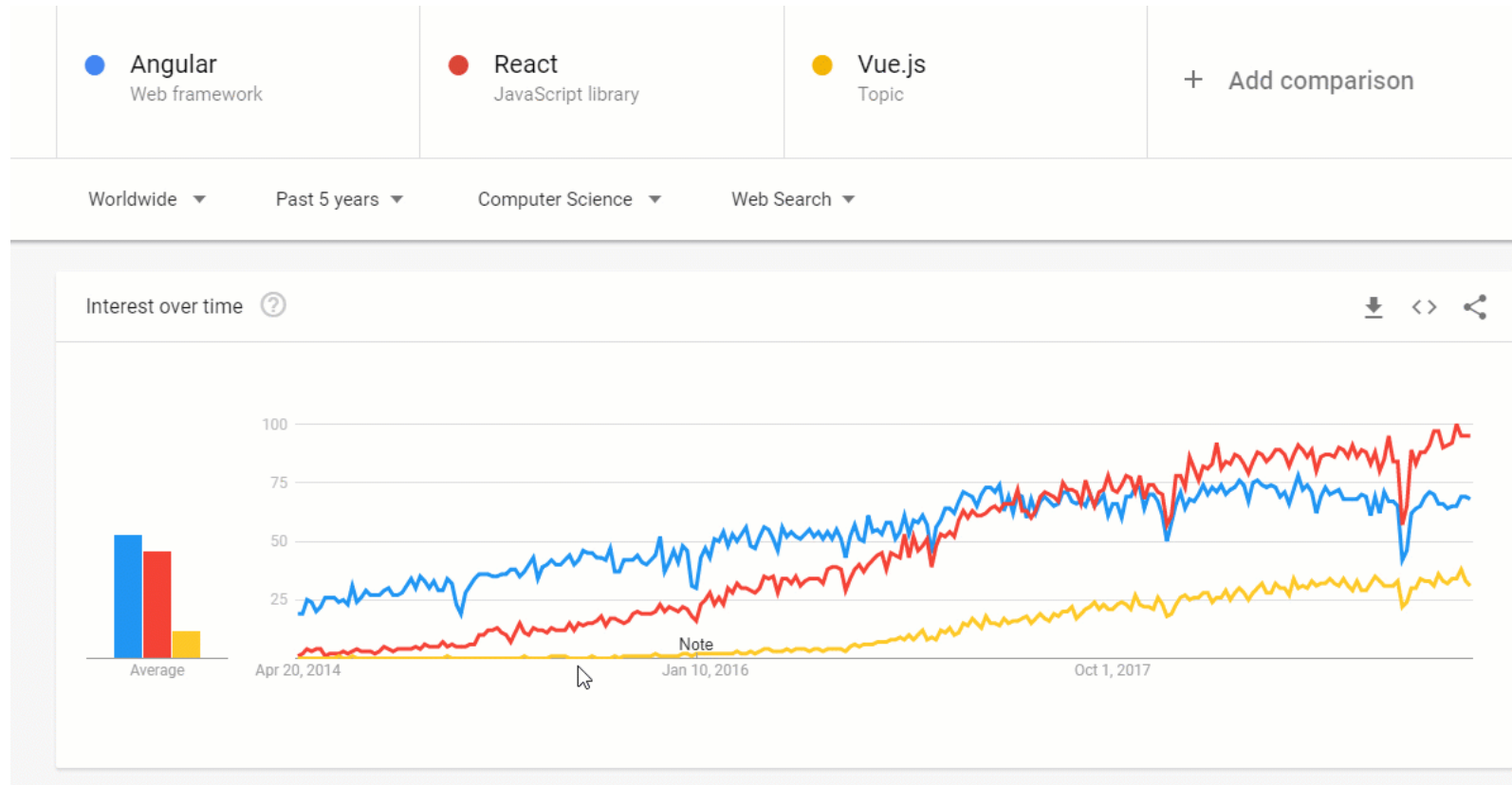
- Адаптивность
- Масштабирование
- Крошечный размер
- Подробная документация

# React

- React Virtual DOM
- Redux
- Привязка данных является односторонней
- Поддержка рендеринга на стороне сервера



# Статистика

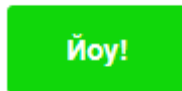


# Button.vue

```
<template>
  <component :is="type" :href="href" :type="submit" :class="['button', size, state, variation]">
    <slot/>
  </component>
</template>
```

ВЫЗОВ:

```
<Button variation="green" size="default">Йой!</Button>
```



# Button.js

```
export default {
  name: "Button",
  status: "prototype",
  props: {
    type: {
      type: String,
      default: "button",
      validator: value => {
        return value.match(/(button|a)/)
      },
    },
    size: {
      type: String,
      default: "medium",
      validator: value => {
        return value.match(/(small|medium|large)/)
      },
    },
    href: {
      type: String,
      default: null,
    },
    submit: {
      type: String,
      default: null,
      validator: value => {
        return value.match(/(null|submit)/)
      },
    },
  },
}
```

# КОМПОЗИЦИЯ

```
<Button variation="green" size="default">  
  <Icon name="blabla" size="default" />  
  Йой!  
</Button>
```







## WEB COMPONENTS

### TEMPLATES

```
<template id="">  
</template>
```

### SHADOW DOM

```
div  
  #document-fragment  
  span
```

### HTML IMPORTS

```
<link rel="import"  
      href="part.html">
```

### CUSTOM ELEMENTS

```
<my-elem>  
</my-elem>
```

## Browser support



 HTML TEMPLATES

 STABLE

 STABLE

 STABLE

 STABLE

 STABLE

 CUSTOM ELEMENTS

 STABLE


 STABLE

 STABLE

 STABLE

 POLYFILL

 DEVELOPING

 SHADOW DOM

 STABLE

 STABLE

 STABLE

 STABLE

 POLYFILL

 DEVELOPING

 ES MODULES

 STABLE

 STABLE

 STABLE

 STABLE

 STABLE

# custom elements

JavaScript API для связывания вашего тега с соответствующим ему классом.

## HTMLTemplateElement

специальный тег `<template>`, предназначенный для хранения шаблонов.

## shadow DOM

Способ инкапсуляции внутреннего содержания тега.



# Button.js

```
class VButton extends HTMLElement {}
```

```
customElements.define("v-button", VButton);
```

```
<audio controls src="kings-speech.wav"></audio>
```



Содержимое Shadow DOM тега <audio> в DevTools:

The screenshot shows the DevTools Elements panel with the following structure:

```
Q Elements Network Sources Timeline Profiles Resources Audits Console
  <html>
    <head>...</head>
    <body>
      <div class="center">
        <audio controls src="http://ccmixter.org/content/flatwound/flatwound - The Long Goodbye.mp3">
          #shadow-root (user-agent)
            <div>
              <div>
                <div>
                  <input type="button">
                  <input type="range" step="any" max="134.506">
                    <div style="display: none;">0:00</div>
                    <div>2:14</div>
                  <input type="button">
                  <input type="range" step="any" max="1" style>
                  <input type="button" style="display: none;">
                  <input type="button" style="display: none;">
                </div>
              </div>
            </div>
          </div>
        </audio>
      </div>
    </body>
  </html>
```

At the bottom of the Elements panel, the breadcrumb path is: html > body > div.center > audio > #shadow-root.

```
<div class="shadow-host">  
  ЭТОТ ТЕКСТ ПОЛЬЗОВАТЕЛЬ НЕ УВИДИТ.  
</div>
```

```
<script>  
  var shadowHost = document.querySelector('.shadow-host');  
  var shadowRoot = shadowHost.createShadowRoot();  
  shadowRoot.innerHTML = 'Он увидит этот текст.'  
</script>
```

PREVIEW

Он увидит этот текст.

```
<template id="color-green">
  <style> div { background-color: green; } </style>
  <div>зелёный</div>
</template>

<div class="shadow-host"></div>

<script>
  var host = document.querySelector('.shadow-host'),
      template = document.querySelector('#color-green'),
      shadow = host.createShadowRoot();
  shadow.appendChild(template.content);
</script>
```

PREVIEW

зелёный

```
shadow.resetStyleInheritance = true;
```

# lifecycle

- `created` — создан инстанс элемента Адаптивность
- `attached` — элемент вставлен в DOM
- `detached` — элемент удалён из DOM
- `attributeChanged` — атрибут элемента добавлен, удалён или изменён

Импорты — простое API, которому давно место в браузерах. Они дают возможность вставлять в документ фрагменты разметки из других файлов.

```
<link rel="import" href="widget.html">
<script>
  var link = document.querySelector('link[rel="import"]');
  // Доступ к импортированному документу происходит с помощью
  // свойства // *import*.
  var importedContent = link.import;
  importedContent.querySelector('article');
</script>
```

# Button + Icon

```
<dom-module id="custom-button">
  <template>
    <div class="button">
      <iron-icon
        class="icon"
        id="icon"
        icon="[[icon]]"
        data-no-content$="[[!_hasContent]]"
        hidden$="[[!icon]]">
      </iron-icon>
      <slot id="content"></slot>
    </div>
  </template>
</dom-module>
```



# Button + Icon

```
class SimpleButton extends Polymer.GestureEventListeners(Polymer.Element) {
  static get is() {
    return 'custom-button';
  }
  static get properties() {
    return {
      icon: {
        type: String,
        observer: '_checkForContent',
        value: ''
      }
    };
  }
  _checkForContent() {
    const hasContent = this.$.content.assignedNodes().length,
          { icon } = this.$;

    if (!hasContent) {
      icon.setAttribute(ICON_CONTENT_ATTR, true);
    } else {
      icon.removeAttribute(ICON_CONTENT_ATTR);
    }
  }
}
```

# Installation & usage

```
<link rel="import" href="/simple-button/simple-button.html">
```

```
<custom-button class="button">Я кнопка</custom-button>
```

# Работа с веб-компонентами



# Bit.dev


### Positions

Right

Top

Bottom

Left




PRIMEREACT /  
**tooltip v3.1.0**

Tooltip functionality is integrated within various PrimeReact...

● 🔄 MIT ❤️

ACTIVE DISABLED ACTIVE




MATERIAL-UI /  
**tab v3.9.2**

Tabs make it easy to explore and switch between different vi...

● 🔄 MIT ❤️

Left Middle Right




REACT-BOOTSTRAP /  
**button-group v1.0.0**

Sets the size for all Buttons in the group.

● 🔄 MIT ❤️

HOVER OR TOUCH FOCUS OR TOUCH HOVER CLICK




MATERIAL-UI /  
**tooltip v3.9.2**

Tooltips display informative text when users hover over, foc...

● 🔄 MIT ❤️

Name	Location	Date	Percent Complete	Price
Alan				\$0.00
Bryan	Fort Collins	5/23/2018	<div style="width: 100%;"></div>	\$12.34
Chris	Provo, Utah	9/9/2018	<div style="width: 100%;"></div>	\$23.45
Eric	Provo, Utah	8/22/2018	<div style="width: 100%;"></div>	\$14.56
Sheng	Fort Collins	8/23/2018	<div style="width: 100%;"></div>	\$12.34
Jet	Provo, Utah	9/6/2018	<div style="width: 100%;"></div>	\$14.56
Michael	Denver	5/15/2018	<div style="width: 100%;"></div>	\$12.34
Therly	San Francisco	5/23/2018	<div style="width: 100%;"></div>	\$23.45
Total				\$123.04




GROMMET /  
**data-table v2.5.5**

A data driven table.

● 🔄 APACHE ❤️

### Fits to Container





SEMANTIC-UI-REACT /  
**button v0.85.0**

A Button indicates a possible user action

● 🔄 MIT ❤️

Combo Bar Line Chart



PRIMEREACT /  
**chart v3.1.0**

Chart components are based on Charts.js, an open source HTML...

● 🔄 MIT ❤️




SEMANTIC-UI-REACT /  
**dimmer v0.85.0**

A dimmer hides distractions to focus attention on particular...

● 🔄 MIT ❤️

### Basic



PRIMEREACT /  
**inputtext v3.1.0**

InputText is an extension to standard input element with the...

● 🔄 MIT ❤️

Username Search Submit



REACT-BOOTSTRAP /  
**navbar v1.0.0**

The general visual variant a the Navbar. Use in combination ...

● 🔄 MIT ❤️

# Bit.dev

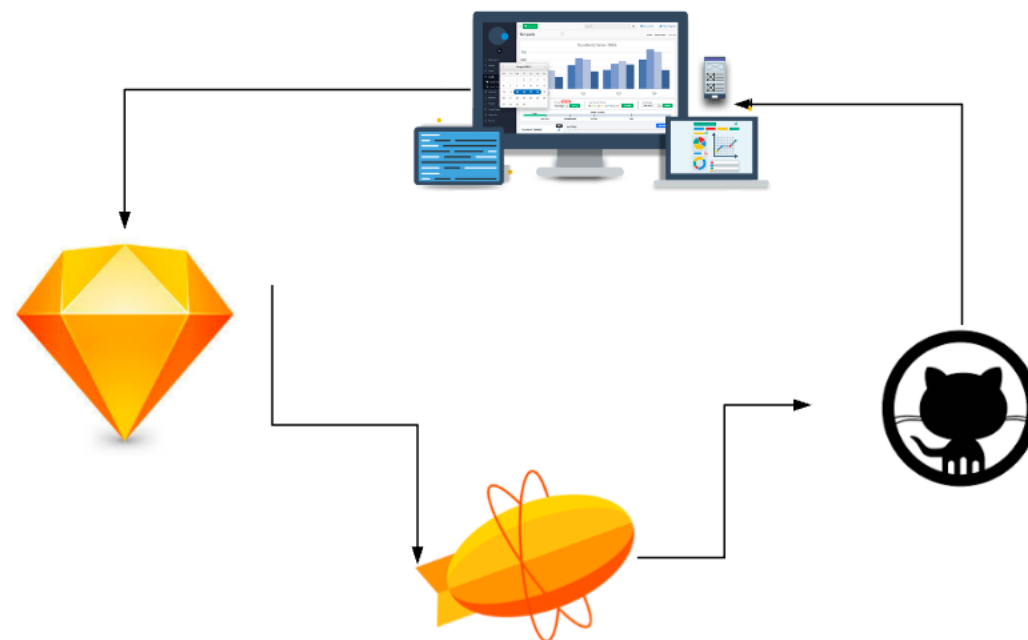
Component Example



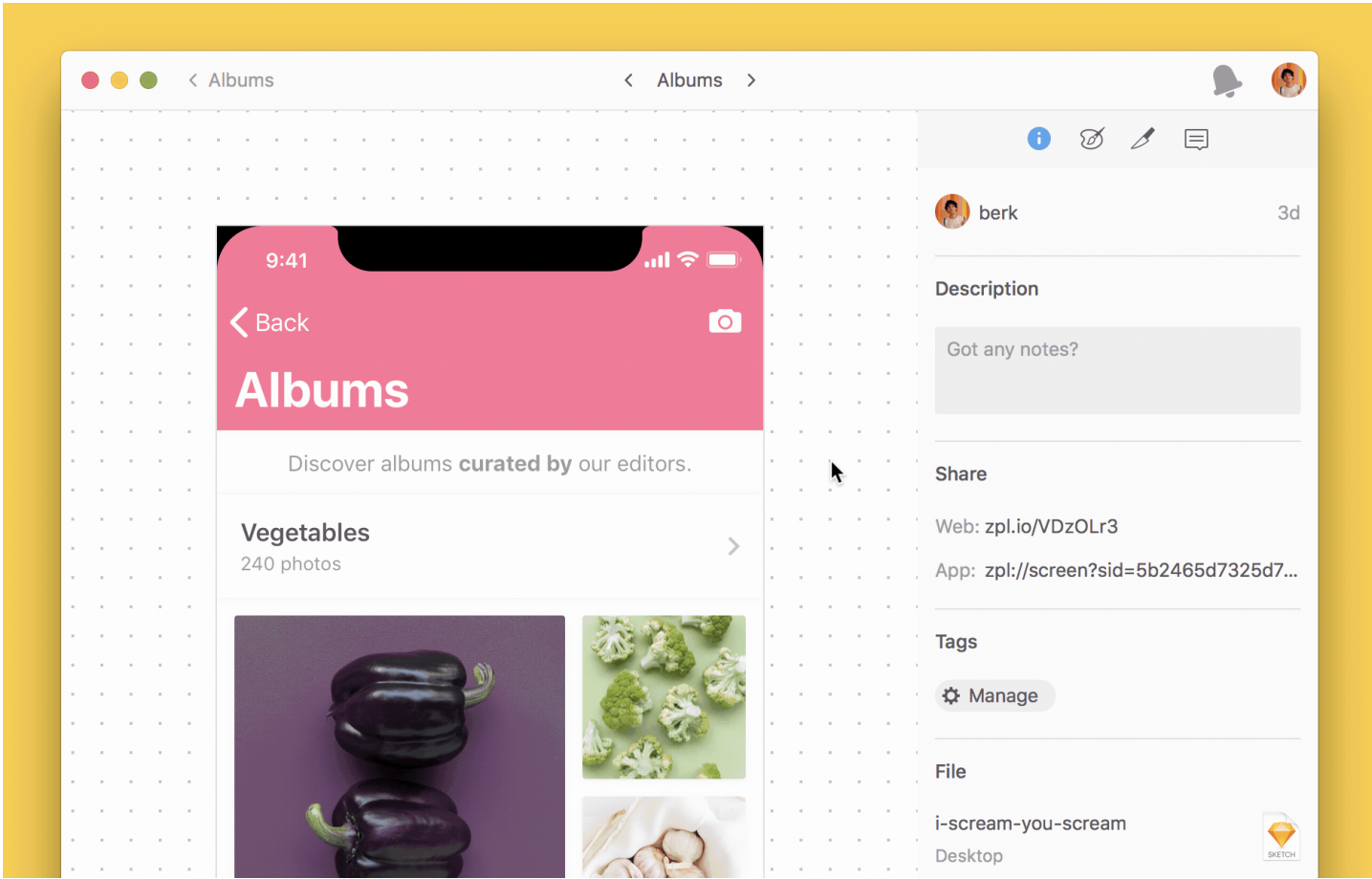
```
1 import React from 'react';
2 import PacmanLoader from '@bit/davidhu2000.react-spinners.pacman-
3
4
5 export default (
6   <PacmanLoader
7     size='90'
8     color='#6b5ce7'
9     css={{width:"260px !important", height:"130px !important"}}
10  />
11 )
12
```



# Инструменты для дизайнеров



# Zeplin



Start from  
mm-dd-yyyy

End on  
mm-dd-yyyy

Travelers  
2 Adults, 2 Roo **276px**

Controls / Button / Primary

Book a trip

**44px**



width: 276px

height: 44px

Radius: 6px

### Component

Controls / Button / Primary

### Button

src/components/Button.jsx

Default button used across the application.  
Set `primary` to display in the brand accent color.

```
<Button  
  primary={Type.bool}  
  disabled={Type.bool}  
  onClick={Type.func}>  
  Text  
</Button>
```

Open in Storybook

Open in GitHub

Open in Visual Studio Code

### Colors

#f07c0b

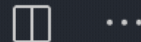


No notes yet





{} components.json ×



```
.zeplin > {} components.json > [ ] components > {} 0 > [ ] zeplinNames > abc 0
```

```
1  {
2      Add project
3      "projects": [
4          "5db0639273cb5206e797c9ad"
5      ],
6      Add styleguide
7      "styleguides": [
8          "5db0639273cb5206e797c9ae"
9      ],
10     Add component
11     "components": [
12         {
13             "path": "src/components/Button.jsx",
14             Add Zeplin component
15             "zeplinNames": [
16                 "Controls / Button / Primary"
17             ]
18         }
19     ]
20 }
```

# Вопросы?

- Компонент
- Фреймворки Angular, Vue, React
- Web-components
- Bit.dev
- Zeplin

