Web Clients

Sabina Borbély 27.04.2020

Einbindung von SVG Signature Sketch in React

2 alternative Implementierungen:

- React Component als Wrapper um ein Custom Element
- React Component als Wrapper um SVG

SVG Signature Sketch in React

wiederverwendbare Komponenten

Bei SVG Signature bieten sich die folgenden Operationen an:

- Änderung von **stroke** und **stroke-width** zur Beeinflussung von Linien
- Änderung von fill zur Veränderung der Füllfarbe
- Änderung von opacity zur Veränderung der Durchsichtigkeit eines Elements
- Veränderung von Position, Größe und Rotation mittels transform

Custom Element Attributes / React Component props:

- stroke fill
- stroke-width
- opacity
- background
- width, height

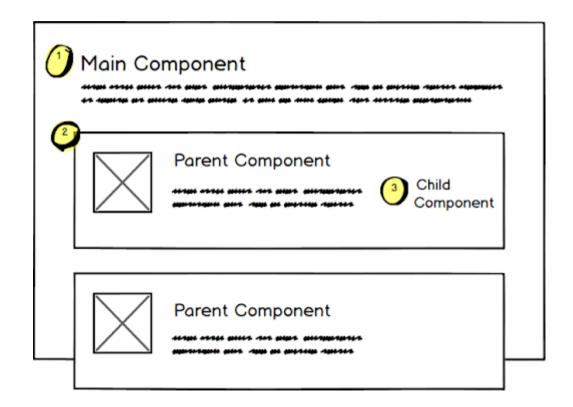
WEB COMPONENTS

Shadow Dom
HTML Templates
Observed Attributes
Lifecycle methods
Custom elements
Register the custom tag

REACT

Functional Component Class Component Props - PropTypes Lifecycle methods Export / Import module

React Components



Web Components

```
export default class Signature extends HTMLElement {
 static get observedAttributes() {
   super();
    this.shadowObj = this.attachShadow({ mode: 'open' });
  connectedCallback() {
   this.init();
    this.shadowObj.querySelector('#signature')
      .addEventListener('click', (e) => this.handleClick(
```

Using Web Components in React

```
componentDidMount() {
   if (!customElements.get('signature-element')) {
     customElements.define('signature-
element', Signature);
   }
}
```

```
constructor(props) {
 super(props);
  this.pathInfo = [
    { path: '' },
    { path: '' }];
  this.pathORef = React.createRef();
  this.path1Ref = React.createRef();
  this.path2Ref = React.createRef();
  this.signatureRef = React.createRef();
    "--stroke-opacity": props.strokeopacity,
    "--stroke-color": props.fillcolor,
   "--stroke-width": props.strokewidth,
    "--bkg-color": props.bkgcolor,
    "--signature-height": props.signatureheight
```

React Class Component - SVG SignatureR2.js

```
componentDidMount() {
    this.pathInfo = [
      { path: this.pathORef.current },
      { path: this.path1Ref.current },
      { path: this.path2Ref.current },
    this.pathInfo.forEach(it => {
      it.total = it.path.getTotalLength();
      it.path.style.strokeDasharray = it.total; // dash w
      it.path.style.strokeDashoffset = it.total; // show
    });
    this.updateCSSCustomProperties(this.props);
    this.signatureRef.current.addEventListener('click', (
e) => this.handleClick());
```

React Class Component - SVG SignatureR2.js

```
render() {
      <div className="SignatureR2" style={this.theme} >
        <div id="signature" ref={this.signatureRef} >
          <svg viewBox="0 -</pre>
2 43 62" fill="none" xmlns="http://www.w3.org/2000/svg">
            <path id="path0" ref={this.path0Ref} d="M18 12V45" stroke="</pre>
            <path id="path1" ref={this.path1Ref} d="M35 4C35 4 19.5 24</pre>
```