React

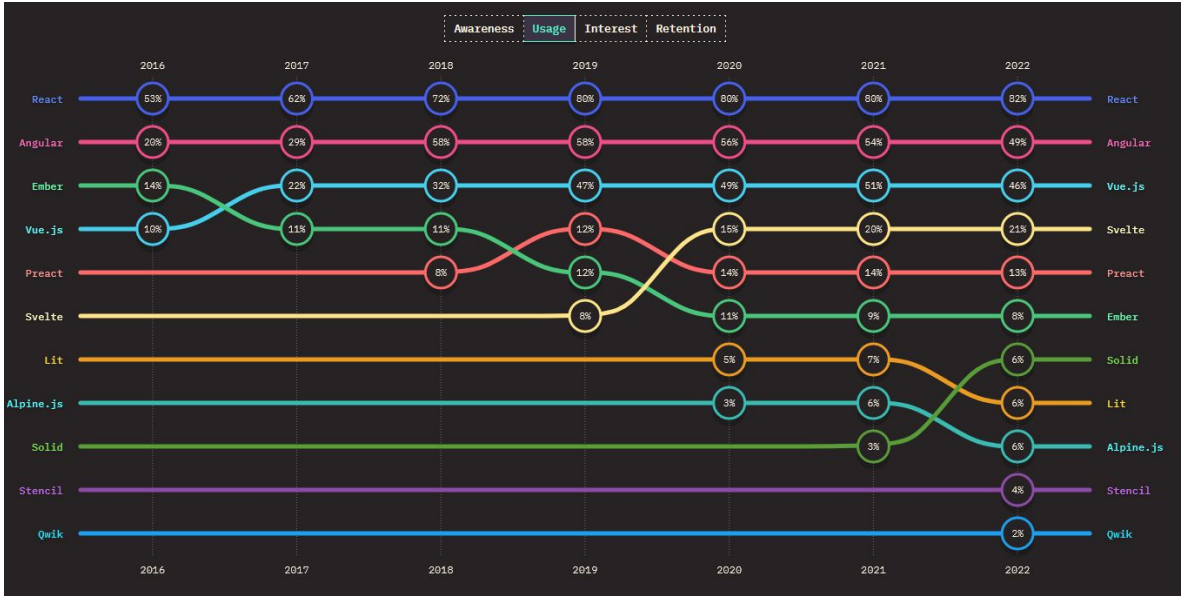
Agnieszka Dańda, Michał Kuś

Czym jest React?

- React to efektywna i elastyczna biblioteka JavaScript służąca do budowania interfejsów użytkownika.
- Została stworzona przez Jordana Walke’a i jest utrzymywana przez Facebook oraz społeczność open source.
- Umożliwia budowanie skomplikowanych interfejsów użytkownika z podziałem na części.

Firmy które używają Reacta

Statystyki i społeczność



Źródła:

- State of Javascript
- Linkedin
- Stackoverflow
- JustJoinIT
- NoFluffJobs

Zastosowanie

- Web Development
- Mobile Development
- Progressive Web Apps (PWA)
- Enterprise Applications

JSX/TSX

- JSX to składnia rozszerzająca JavaScript, która pozwala na pisanie kodu, który wygląda jak HTML
- TSX to wariant JSX, który jest używany z TypeScript, dodając bezpieczeństwo typów do projektów React. Dzięki TSX, programiści mogą korzystać z zalet TypeScript, takich jak statyczne typowanie i lepsze narzędzia do refaktoryzacji.
- Korzyści: większa prostota

XML (eXtensible Markup Language)

XML został zaprojektowany w celu przechowywania i przesyłania danych w sposób prosty i strukturalny.

Służy do przechowywania i przesyłania danych w formie strukturalnej.

Wykorzystywany w konfiguracjach aplikacji, wymianie danych między systemami oraz web services.

Komponenty w React

W React istnieją dwa główne typy komponentów:

Komponenty Klasowe: Starsza forma komponentów, które używają ES6 classes.

Komponenty Funkcyjne: Nowoczesne, bardziej popularne podejście wykorzystujące hooks (np. useState, useEffect) do zarządzania stanem i efektami.

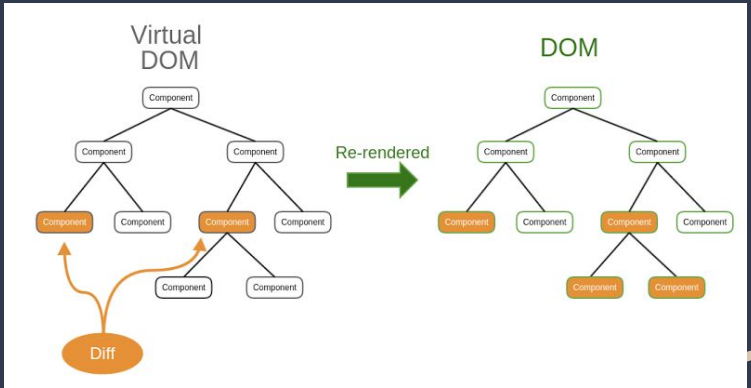
Komponenty w React

```
class Employee extends React.Component {

  render() {
    return (
      <div>
        <h2>Employee Details...</h2>
        <p>
          <label>Name : <b>{this.props.Name}</b></label>
        </p>
        <Department Name={this.props.DeptName} />
      </div>
    )
  }
}

class Department extends React.Component {

  render() {
    return (
      <div>
        <h2>Department Details...</h2>
        <p>
          <label>Name : <b>{this.props.Name}</b></label>
        </p>
      </div>
    )
  }
}

const element = <Employee Name="Pragim" DeptName="Dev" />;

ReactDOM.render(element,document.getElementById("root"));
```

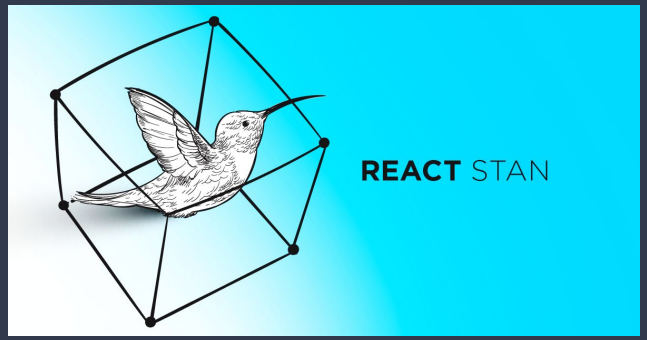
```
function UserList() {
  const [users, setUsers] = useState<User[]>([]);
  const [error, setError] = useState<any>(null);

  useEffect(() => {
    async function fetchData() {
      try {
        const data = await fetchUsers();
        setUsers(data);
      } catch (error) {
        setError(error);
      }
    }
    fetchData();
  }, []);

  if (error) {
    return <div>Error: {error}</div>;
  } return (
    <div>
      {users.map((user) => (
        <div key={user.id}>
          <h2>{user.name}</h2>
          <p>{user.email}</p>
          <UserCard descrption={user.description}/>
        </div>
      ))}
    </div>
  );
}
```

```
const UserCard = ({ description }: { description: string }) => {
  const { isOpen, onOpen, onClose } = useDisclosure()

  return (
    <>
      <Modal blockScrollOnMount={false} isOpen={isOpen} onClose={onClose}>
        <ModalOverlay />
        <ModalContent>
          <ModalHeader>Card Title</ModalHeader>
          <ModalCloseButton />
          <ModalBody>
            <Text fontWeight='bold' mb='1rem'>
              { description }
            </Text>
          </ModalBody>

          <ModalFooter>
            <Button colorScheme='blue' mr={3} onClick={onClose}>
              Close
            </Button>
            <Button variant='ghost'>Secondary Action</Button>
          </ModalFooter>
        </ModalContent>
      </Modal>
    </>
  )
}

export default UserCard;
```

Virtual DOM



- Virtual DOM (VDOM) to koncepcja programowania używana w React do poprawy wydajności aplikacji.
- Kiedy stan komponentu się zmienia, React tworzy nowe drzewo VDOM, które porównuje z poprzednim. Następnie oblicza najefektywniejszy sposób na zaktualizowanie rzeczywistego DOM na podstawie tej różnicy.
- Korzyści: szybkie aktualizacje UI bez potrzeby przeładowywania całej strony



- Stan odnosi się do danych, które determinują zachowanie komponentu i jak jest on renderowany.
- Może być aktualizowany, co prowadzi do ponownego renderowania komponentu.
- Przechowuje informacje takie jak aktualnie zalogowany użytkownik, czy wartość pola formularza.

Wady i zalety

Wady:

- Brak architektury
- Złożoność w dużych aplikacjach
- Szybkie tempo rozwoju
- Brak spójnej dokumentacji

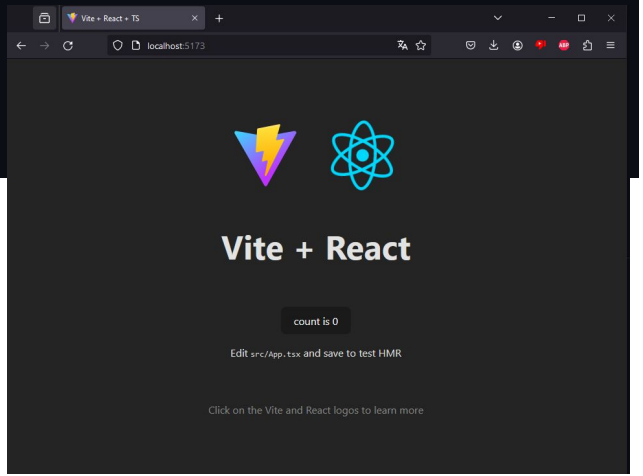
Zalety:

- Wydajność
- Elastyczność
- Duża społeczność i wsparcie
- Łatwość nauki
- Wiele możliwości dla użytkowników
- Szerokie zastosowanie
- Efektywność
- Skalowalność
- Wieloplatformowość
- Prostota w zastosowaniu

Tworzenie projektu - React

Vite to nowoczesne narzędzie deweloperskie, które umożliwia szybkie tworzenie aplikacji w wybranej technologii, szczególnie w środowisku React. Dzięki wydajnemu systemowi budowania opartemu na technologii ESM (ECMAScript Modules), Vite oferuje natychmiastowe ładowanie modułów oraz dynamiczne odświeżanie przeglądarki, co przyspiesza proces rozwoju i ułatwia iteracyjną pracę nad projektem.

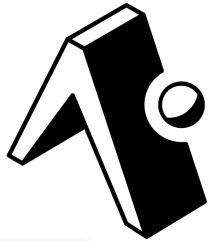
```
PS D:\Repository\React> npm create vite
Need to install the following packages:
  create-vite@5.2.3
Ok to proceed? (y) y
√ Project name: ... my-project
√ Select a framework: » React
√ Select a variant: » TypeScript

Scaffolding project in D:\Repository\React\my-project...

Done. Now run:

  cd my-project
  npm install
  npm run dev
```

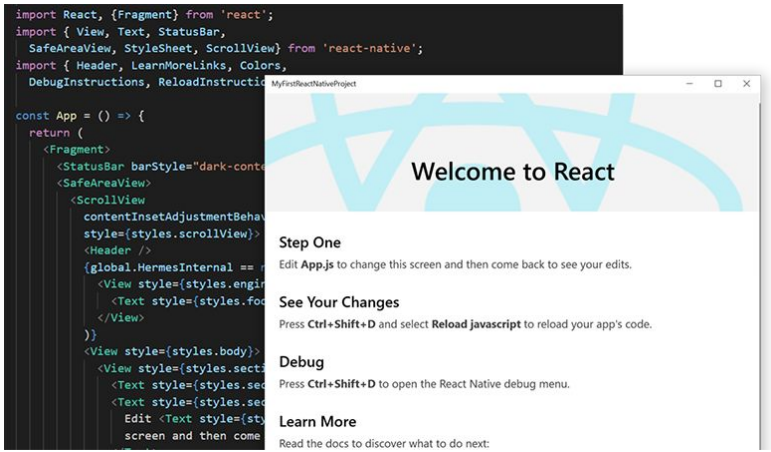


Tworzenie projektu - React Native

Kilka sposób instalacji i wykorzystania React Native

- React Native with Expo
- React Native CLI with XCode/Android Emulator
- React Native Desktop for Windows
- React Native Desktop for macOS





Różnice między Reactem a Angularem

React:

- Biblioteka, dzięki której budujemy aplikacje składając ją z innych modułów.
- JSX/TSX
- Brak określonej struktury
- One-way binding
- Virtual DOM

Angular

- Pełnoprawny duży framework oparty na MVC. Stosowany w dużych projektach.
- Typescript
- Określona struktura narzucona przez Angulara
- Two-way data binding
- Change Detection

Źródła

https://react.dev/

https://2022.stateofjs.com/en-US

https://reactnative.dev/

https://stackoverflow.com/