FRONTEND. REACT. ЛЕКЦІЯ 1

ЖИТОМИРСЬКА ПОЛІТЕХНІКА
ДЕРЖАВНИЙ УНІВЕРСИТЕТ

BEER CS

ХТО Я

- Савіцький Роман
- Infopulse frontend service line lead
- BeerJS Zhytomyr co-founder
- OdessaJS / Frontend://zt
- Суддя DEV Challenge
- Спікер

ФРОНТЕНД

Фреймворк. Ангуляр

Фреймворк. Ангуляр

Бібліотека. Реакт

Фреймворк. Ангуляр

Бібліотека. Реакт

Бібліотека. Вью

ФРОНТЕНД

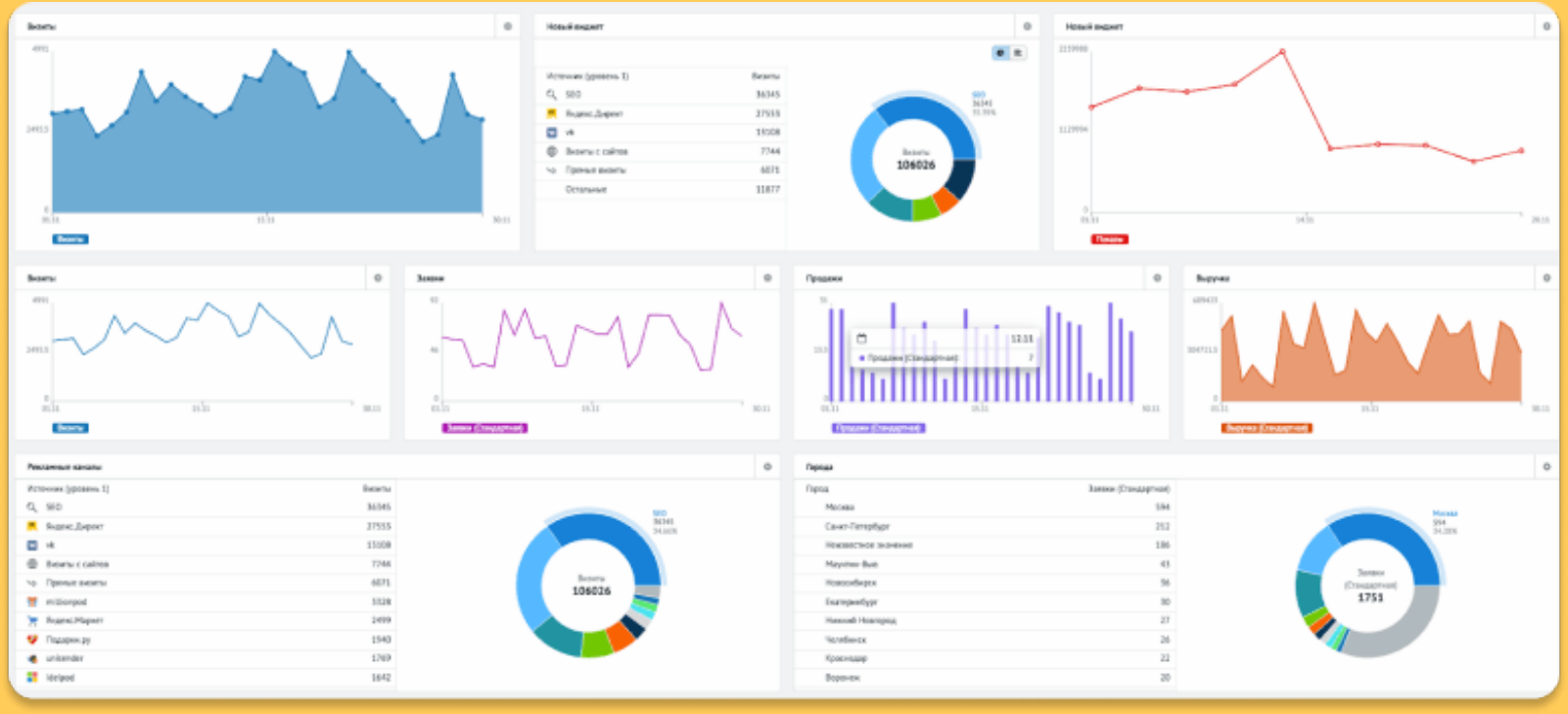
Фреймворк. Ангуляр

Бібліотека. Реакт

Бібліотека. Вью

Підхід. Свелте

ЯКУ ПРОБЛЕМУ ВИРІШУЮТЬ ФРОНТЕНД ФРЕЙМВОРКИ?



ЯКУ ПРОБЛЕМУ ВИРІШУЮТЬ ФРОНТЕНД ФРЕЙМВОРКИ

ЯКУ ПРОБЛЕМУ ВИРІШУЮТЬ ФРОНТЕНД ФРЕЙМВОРКИ

- Проблема контролю та синхронізації данних та їх відображення

ЯКУ ПРОБЛЕМУ ВИРІШУЮТЬ ФРОНТЕНД ФРЕЙМВОРКИ

- Проблема контролю та синхронізації данних та їх відображення
- Швидкий рендерінг DOM (бажано 16fps)

ЩО ТАКЕ РЕАКТ?

ЩО ТАКЕ РЕАКТ?

Трансформація данних в розмітку

ЩО ТАКЕ РЕАКТ?

Трансформація данних в розмітку

Перемальовування данних

ЩО ТАКЕ РЕАКТ?

Трансформація данних в розмітку

Перемальовування данних

Open source бібліотека для створення веб інтерфейсів

ОФІЦІЙНА ДОКУМЕНТАЦІЯ

real time.

LIVE JSX EDITOR ☑ JSX?

```
class MarkdownEditor extends React.Component {
  constructor(props) {
    super(props);
    this.md = new Remarkable();
    this.handleChange =
this.handleChange.bind(this);
    this.state = { value: 'Hello, **world**!' };
  }

  handleChange(e) {
    this.setState({ value: e.target.value });
  }

  getRawMarkup() {
    return { __html:
this.md.render(this.state.value) };
```

RESULT

Input

Enter some markdown

Hello, **world**!

Output

Hello, **world**!

Get Started Take the Tutorial >

ДЕ ВИКОРИСТОВУЄТЬСЯ РЕАКТ

ДЕ ВИКОРИСТОВУЄТЬСЯ РЕАКТ

- SSR server

ДЕ ВИКОРИСТОВУЄТЬСЯ РЕАКТ

- SSR server
- react web

ДЕ ВИКОРИСТОВУЄТЬСЯ РЕАКТ

- SSR server
- react web
- react-native ios, android apps

ПРИНЦИПИ РЕАКТ

ПРИНЦИПИ РЕАКТ

Декларативний

ПРИНЦИПИ РЕАКТ

Декларативний

Контроль данних, а не розмітки

ПРИНЦИПИ РЕАКТ

Декларативний

Контроль данних, а не розмітки

Базується на компонентах

УСТАНОВКА I СТАРТ

NODE

Node.js – середовище виконання javascript на основі рушія V8, розробленого Google.

Розробник – Раян Дал

2009 р

ГОЛОВНА | ПРО ПРОЕКТ | ЗАВАНТАЖЕННЯ | ДОКУМЕНТАЦІЯ | ПРИЄДНАТИСЬ | БЕЗПЕКА | НОВИНИ | CERTIFICATION

Node.js® — це JavaScript–оточення побудоване на JavaScript–рушієві Chrome V8.

New security releases now available for 18.x, 16.x, and 14.x release lines

Завантажити для macOS (x64)

16.17.1 LTS

Рекомендовано для більшості

Інші завантаження | Список змін | Документація

18.9.1 Поточна

Найновіші можливості

Інші завантаження | Список змін | Документація

Або подивіться на графік LTS

NODEJS

NODEJS

- npm (Node Package Manager) – менеджер, що допомагає встановлювати пакети та їх залежності

NODEJS

- npm (Node Package Manager) – менеджер, що допомагає встановлювати пакети та їх залежності
- npx (Node Package Executor) – менеджер, що допомагає встановлювати пакети та запускає їх

ДОДАТИ РЕАКТ ЧЕРЕЗ ПОСИЛАННЯ

- Add React in One Minute
- Optional: Try React with JSX (no bundler necessary!)

Add React in One Minute

In this section, we will show how to add a React component to an existing HTML page. You can follow along with your own website, or create an empty HTML file to practice.

There will be no complicated tools or install requirements — **to complete this section, you only need an internet connection, and a minute of your time.**

Optional: Download the full example (2KB zipped)

Step 1: Add a DOM Container to the HTML

INSTALLATION

Getting Started

Add React to a Website

Create a New React App

CDN Links

Release Channels

MAIN CONCEPTS

ADVANCED GUIDES

API REFERENCE

HOOKS

TESTING

ВСТАНОВЛЕННЯ РЕАКТ ЗА ДОПОМОГОЮ NPX CREATE-REACT-APP

npm пакет від компанії Meta

On this page

Getting Started

Create React App is an officially supported way to create single-page React applications. It offers a modern build setup with no configuration.

Quick Start

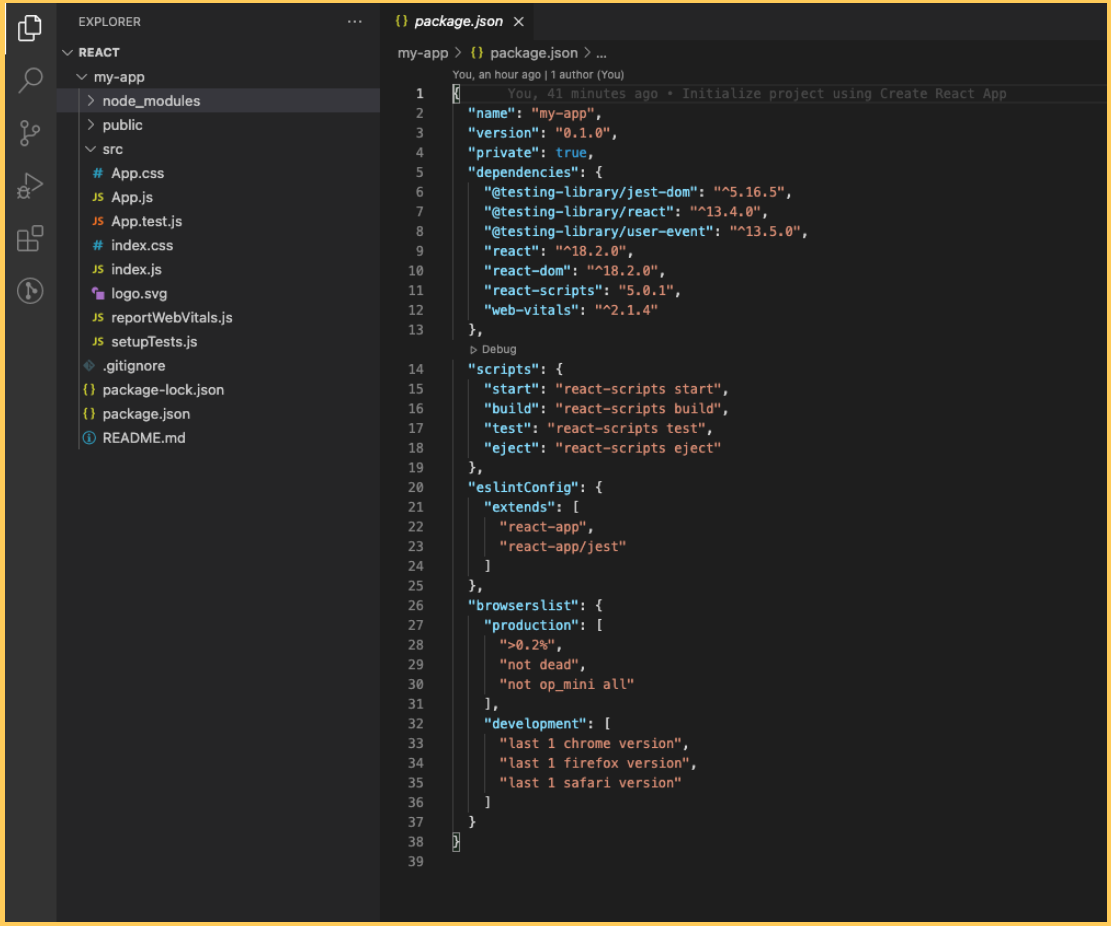
```
npx create-react-app my-app
cd my-app
npm start
```

> If you've previously installed `create-react-app` globally via `npm install -g create-react-app`, we recommend you uninstall the package using `npm uninstall -g create-react-app` or `yarn global remove create-react-app` to ensure that `npx` always uses the latest version.

(npx comes with npm 5.2+ and higher, see instructions for older npm versions)

Then open http://localhost:3000/ to see your app.

БАЗОВА СТРУКТУРА ПРОЕКТУ. PACKAGE.JSON



БАЗОВІ КОМАНДИ

БАЗОВІ КОМАНДИ

```
npm run start
```

БАЗОВІ КОМАНДИ

```
npm run start
```

```
npm run build
```

БАЗОВІ КОМАНДИ

```
npm run start
```

```
npm run build
```

```
npm run test
```

БАЗОВІ КОМАНДИ

```
npm run start
```

```
npm run build
```

```
npm run test
```

```
npm run eject
```

РЕНДЕРИНГ

РЕНДЕРИНГ

- DOM (Document Object Model)

РЕНДЕРИНГ

- DOM (Document Object Model)
- DOM документ є деревом

РЕНДЕРИНГ

- DOM (Document Object Model)
- DOM документ є деревом
- Гілки дерева оновлюються повністю

VIRTUAL DOM

VIRTUAL DOM

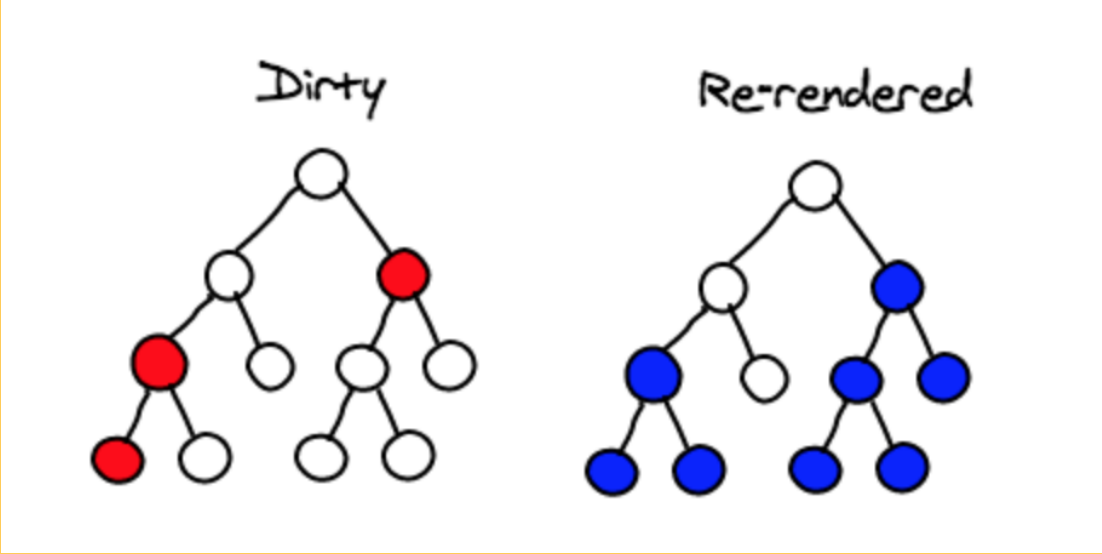
- React-DOM працює з віртуальним DOM

VIRTUAL DOM

- React-DOM працює з віртуальним DOM
- Віртуальний DOM зберігається в пам'яті і синхронізується з DOM

VIRTUAL DOM

- React-DOM працює з віртуальним DOM
- Віртуальний DOM зберігається в пам'яті і синхронізується з DOM
- Процес узгодження (reconciliation)



ЯК РЕАКТ ВИКОРИСТОВУЄ VIRTUAL DOM?

ЯК REAKT ВИКОРИСТОВУЄ VIRTUAL DOM?

- React спостерігає (Observer) за станами компонентів

ЯК REAKT ВИКОРИСТОВУЄ VIRTUAL DOM?

- React спостерігає (Observer) за станами компонентів
- При зміні стану компонента, оновлює VDOM

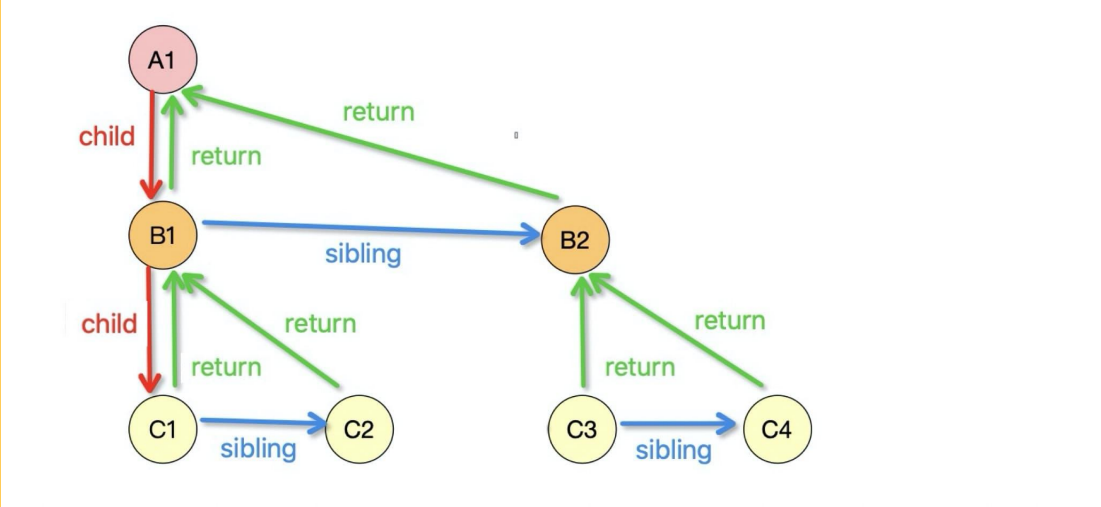
ЯК REAKT ВИКОРИСТОВУЄ VIRTUAL DOM?

- React спостерігає (Observer) за станами компонентів
- При зміні стану компонента, оновлює VDOM
- Порівняння VDOM з попередньою версією (diffing)

ЯК РЕАКТ ВИКОРИСТОВУЄ VIRTUAL DOM?

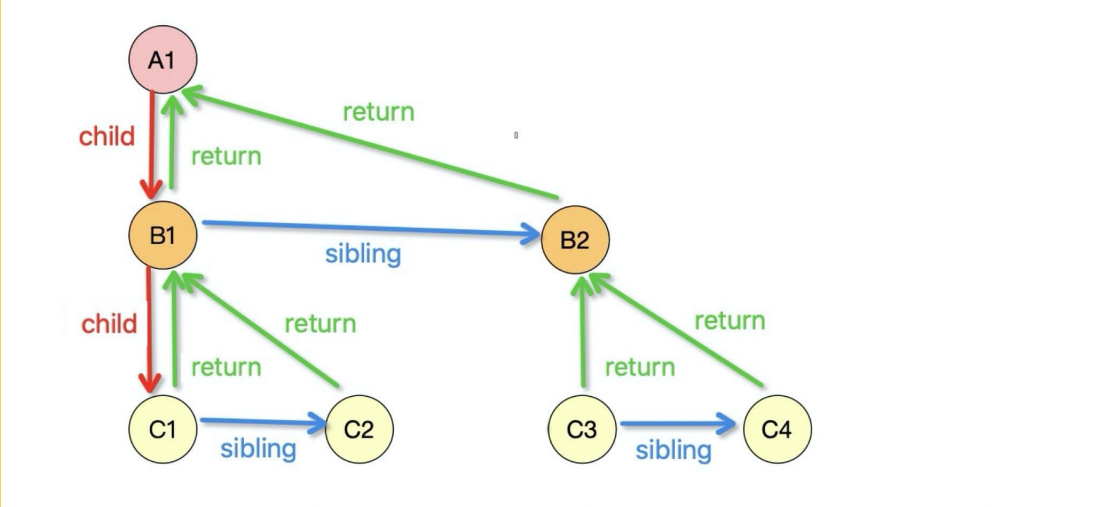
- React спостерігає (Observer) за станами компонентів
- При зміні стану компонента, оновлює VDOM
- Порівняння VDOM з попередньою версією (diffing)
- Реакт переносить оновлення в реальному DOM через механізм пакетного оновлення (batch)

REACT FIBER


A1
child
return
return
B1
sibling
B2
child
return
return
return
return
C1
sibling
C2
C3
sibling
C4

REACT FIBER

- React використовує React Fiber як механізм узгодження



КОМПОНЕНТИ

КОМПОНЕНТИ

- Найменша атомарна одиниця в реакт

КОМПОНЕНТИ

- Найменша атомарна одиниця в реакт
- Розроблені для повторного використання

КОМПОНЕНТИ

- Найменша атомарна одиниця в реакт
- Розроблені для повторного використання
- Містять props, які приймають данні через параметри

КОМПОНЕНТИ

- Найменша атомарна одиниця в реакт
- Розроблені для повторного використання
- Містять props, які приймають данні через параметри
- Описується станом state, що зберігає данні властивостей

КОМПОНЕНТИ

- Найменша атомарна одиниця в реакт
- Розроблені для повторного використання
- Містять props, які приймають данні через параметри
- Описується станом state, що зберігає данні властивостей
- Містять props & state

КОМПОНЕНТИ

- Найменша атомарна одиниця в реакт
- Розроблені для повторного використання
- Містять props, які приймають данні через параметри
- Описується станом state, що зберігає данні властивостей
- Містять props & state
- Функціональні та класові компоненти

ФУНКЦІОНАЛЬНИЙ КОМПОНЕНТ

ФУНКЦІОНАЛЬНИЙ КОМПОНЕНТ

```
function Welcome(props) {
        return <h1>Привет, {props.name}</h1>;
}

function App() {
        return (
                <div>
                        <welcome name="Алиса"></welcome>
                        <welcome name="Базилио"></welcome>
                        <welcome name="Буратино"></welcome>
                </div>
        );
}
```

JSX

Розширення мови Javascript, що дозволяє використовувати HTML-подібний синтаксис.

Вступ до JSX

Розглянемо оголошення змінної:

```
const element = <h1>Hello, world!</h1>;
```

Цей кумедний синтаксис тегів не є ні рядком, ні HTML.

Він має назву JSX, і це розширення синтаксису для JavaScript. Ми рекомендуємо використовувати його в React, щоб описати, як повинен виглядати інтерфейс користувача. JSX може нагадувати мову шаблонів, але з усіма перевагами JavaScript.

ВСТАНОВЛЕННЯ

ОСНОВНІ ПОНЯТТЯ

1. Привіт, світе
2. **Вступ до JSX**
3. Рендеринг елементів
4. Компоненти і пропси
5. Стан та життєвий цикл
6. Обробка подій
7. Умовний рендеринг
8. Списки та ключі
9. Форми
10. Підйом стану
11. Композиція проти наслідування
12. Філософія React

ШО? ШО? АБО ВАШІ ЗАПИТАННЯ