

## Tartalomjegyzék

0., Előfeltétel Node.js telepítése.....	2
1., REACT könyvtár (Library) telepítése – új projekt létrehozása .....	2
1.1., GitHub használata .....	2
2., REACT kiegészítők hozzáadása a projekthez.....	3
2.1., Bootstrap telepítése .....	3
2.1.1., A Bootstrap CSS importálása.....	4
2.1.2., A Bootstrap ikonok telepítése .....	4
2.1.3., A React-Bootstrap komponensek importálása .....	4
2.2., React Router telepítése.....	4
3., Szükségtelen állományok törlése .....	4
4., Alap fájlok (App.js és index.js) beállítása.....	5
5., Adatok tárolása.....	5
6., Saját komponens készítése .....	5

## 0., Előfeltétel Node.js telepítése

---

<https://nodejs.org/en/download/> oldalról letölteni és telepíteni.

Az NPM egy csomagkezelő a Node.js csomagokhoz, modulokhoz. Az NPM a Node.js telepítésekor kerül a számítógépre.

NPM verzió ellenőrzése: **npm -v**

Node verzió ellenőrzése: **node -v**

Visual Studio Code telepítése: <https://code.visualstudio.com/Download>

VS Code ajánlott bővítmények:

1. Simple React Snippets (Burke Holland)  
<https://marketplace.visualstudio.com/items?itemName=burkeholland.simple-react-snippets>  
VAGY
2. ES7+ React/Redux/React-Native snippets (dsznajder)  
<https://marketplace.visualstudio.com/items?itemName=dsznajder.es7-react-js-snippets>  
ÉS
3. Prettier - Code formatter (Prettier)  
<https://marketplace.visualstudio.com/items?itemName=esbenp.prettier-vscode>

## 1., REACT könyvtár (Library) telepítése – új projekt létrehozása

---

NPM (NPX) installerrel:

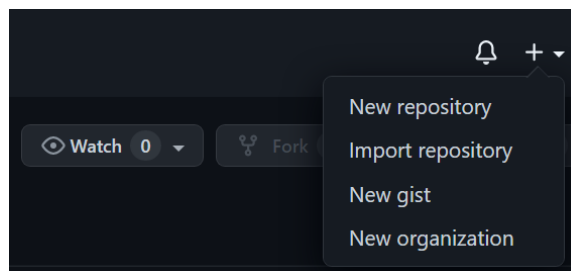
```
npx create-react-app bukizoli-20221102
cd bukizoli-20221102
code . // Visual Studio Code futtatása projekt megnyitásával
npm start // Development server futtatása
127.0.0.1:3000 // Böngészőben erről az útról lehet az oldalt megtekinteni
localhost:3000 // ... vagy erről.
```

### 1.1., GitHub használata

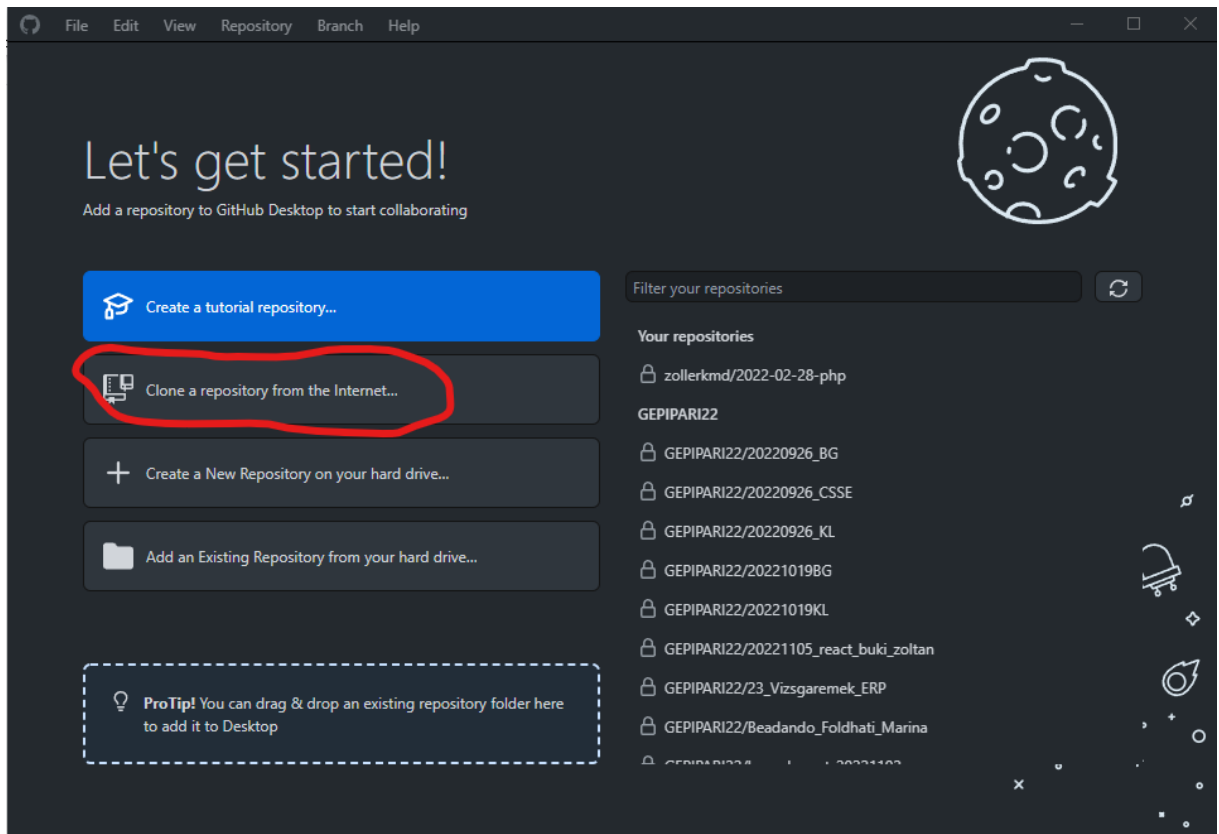
Előfeltétel a GitHub asztali alkalmazás telepítése és abba való bejelentkezés:

<https://desktop.github.com/>

GitHub használata esetén először a github.com oldalon új repository-t kell létrehozni (nagybetűt ne tartalmazzon!):



Majd a GitHub asztali alkalmazásban klónozni kell a fentiek szerint létrehozott repository-t, azaz válasszuk az alkalmazásban a „Clone a repository from the Internet...” lehetőséget:



... és válasszuk ki a felugró ablak listájából a létrehozott repository-t és a „Local path” mezőben a saját számítógépen kiválasztott könyvtárat adjuk meg, ahova a repository kerül, majd kattintsunk a „Clone” gombra.

Most lehet a React projektet létrehozni a local path-ként megadott könyvtárban:

```
npx create-react-app ./  
code . // Visual Studio Code futtatása projekt megnyitásával  
npm start // Development server futtatása  
127.0.0.1:3000 // Böngészőben erről az útról lehet az oldalt megtekinteni  
localhost:3000 // ... vagy erről.
```

GitHub asztali alkalmazásban feltöltés github.com oldalra:

„Commit to **main**”  
„Fetch origin”

## 2., REACT kiegészítők hozzáadása a projekthez

### 2.1., Bootstrap telepítése

A React-Bootstrap (weboldal: <https://react-bootstrap.github.io/>) és Bootstrap (<https://getbootstrap.com/>) hozzáadásának legegyszerűbb módja NPM használatával történik:

```
npm install react-bootstrap bootstrap
```

VAGY

Bootstrap telepítése **npm** segítségével:

```
npm install bootstrap@v5.2.2
```

A @v5.2.2 elhagyható

### 2.1.1., A Bootstrap CSS importálása

A Bootstrap CSS állományt az **index.js** vagy **App.js** fájlba kell importálni:

```
// Importing the Bootstrap CSS
import 'bootstrap/dist/css/bootstrap.min.css';

VAGY
// Importing the Bootstrap CSS
import './bootstrap/dist/css/bootstrap.min.css';
```

Ha nem működne, akkor a Bootstrap oldaláról le kell tölteni a szükséges CSS fájlt:

<https://getbootstrap.com/docs/5.2/getting-started/download/>

A projekt 'src' mappájába be kell másolni a bootstrap.min.css állományt a következő könyvtárszerkezettel:

```
'bootstrap/dist/css/bootstrap.min.css'
```

### 2.1.2., A Bootstrap ikonok telepítése

A Bootstrap ikonokat parancssorból lehet telepíteni a projekthez:

```
npm install bootstrap-icons
```

### 2.1.3., A React-Bootstrap komponensek importálása

A React-Bootstrap hozzáadásának legegyszerűbb módja NPM használatával történik, pl az **App.js** fájlhoz:

```
import Button from 'react-bootstrap/Button';

// tömörebben (több komponens esetén ideálisabb):
import { Button } from 'react-bootstrap';
```

## 2.2., React Router telepítése

React router és react-router-dom telepítése

```
npm install react-router-dom
```

Fentiek egyben is telepíthetők az alábbi paranccsal:

```
npm install react-bootstrap bootstrap bootstrap-icons react-router-dom
```

## 3., Szükségtelen állományok törlése

---

A projektből törölhetők az alábbi állományok:

- App.css
- App.test.js
- index.css
- reportWebVitals.js
- setupTests.js

Az App.js és az index.js állományokból töröljük a fentiekre mutató bejegyzéseket!

## 4., Alap fájlok (App.js és index.js) beállítása

---

Az App.js szükséges tartalma:

Érdemes arrow functionként létrehozni:

rafce →  
(rafce és Tab leütése)

snippettel készen kapjuk a React **App stateless** komponenst:

```
import React from "react";

const App = () => {
  return <div>App</div>;
};

export default App;
```

Az index.js szükséges tartalma:

```
import React from 'react';
import ReactDOM from 'react-dom/client';
import App from './App';
// Importing the Bootstrap CSS
import 'bootstrap/dist/css/bootstrap.min.css';

const root =
ReactDOM.createRoot(document.getElementById('root'));
root.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>
);
```

## 5., Adatok tárolása

---

Az 'src' mappában hozzuk létre a 'utils' nevű könyvtárat, majd abban a **data.js** fájlt, pl az alábbi tartalommal:

```
module.exports = {
}
```

## 6., Saját komponens készítése

---

Komponensek tárolására az 'src' mappában hozzuk létre a 'components' nevű könyvtárat!

Készítsünk egy pl Header komponenst, **Header.jsx** néven az **'src/components//Header/'** mappába:

Megjegyzés: Visual Studio Code programban érdemes a „VS Code ES7+ React/Redux/React-Native/JS snippets” bővítményt (weboldal:

<https://marketplace.visualstudio.com/items?itemName=rodrigovallades.es7-react-js-snippets>)

telepíteni, mert így az

rce →  
(rce és Tab leütése)

snippettel készen kapjuk a React **Stateful** komponenst:

```
import React, { Component } from 'react'

export class Header extends Component {
  render() {
    return (
      <div>
        <p>Header</p>
        <p>{ this.state.first_name }</p>
        <p>{ this.state.last_name }</p>
      </div>
    )
  }
}

export default Header;
```

Illetve 'class' elejére, még a 'render()' előtt az 'rconst →' snippettel be tudjuk állítani a konstruktorát:

```
constructor(props) {
  super(props)

  this.state = {
    first_name : 'Zoltán'
    last_name : 'Büki'
  }
}
```

Így egyben a 'Header' **Stateful** komponens:

```
import React, { Component } from 'react'

export class Header extends Component {
  constructor(props) {
    super(props)

    this.state = {
      first_name : 'Zoltán',
      last_name : 'Büki',
    }
  }
}
```

```

    }

    render() {
      return (
        <div>
          <p>Header komponens</p>
          <p>{this.state.first_name}</p>
          <p>{this.state.last_name}</p>
        </div>
      )
    }
  }

  export default Header;

```

Ugyan ez **Stateless** komponensként, az

rfce →  
(rfce és Tab leütése)

snippettel készen kapjuk a React **Stateless** komponenst:

```

import React from 'react';

function HeaderStateless(props) {
  const adatok = {
    first_name : 'Zoltán',
    last_name : 'Büki'
  }

  return (
    <div>
      <p>Header Stateless komponens</p>
      <p>{adatok.first_name}</p>
      <p>{adatok.last_name}</p>
      <p>{props.atad}</p>
    </div>
  )
}

export default HeaderStateless;

```

A fentiben a 'props' egy átadott érték ('atad'), amit a következőben hívtunk meg, pl az App.js fájlból:

```

import logo from './logo.svg';
//import './App.css';

import React, { Component } from 'react'
import Header from './components/header';

```

```
import HeaderStateless from './components/HeaderStateless';

export class App extends Component {
  render() {
    return (
      <div>
        <p>szia</p>
        <Header />
        <HeaderStateless atad="ez az átadott props"/>
        <HeaderStateless atad="ez egy másik átadott props"/>
      </div>
    )
  }
}

export default App
```

## 6.1., Menü (NavBar) készítése

**App.js** fájlban dolgozunk!

Szükséges import:

```
import React from "react";
import { Container, Nav, Navbar } from "react-bootstrap";
import { BrowserRouter, Routes, Route, Link } from "react-router-dom";
```