

[Sign In](#)[Get started](#)

Published in Level Up Coding

You have **2** free member-only stories left this month.

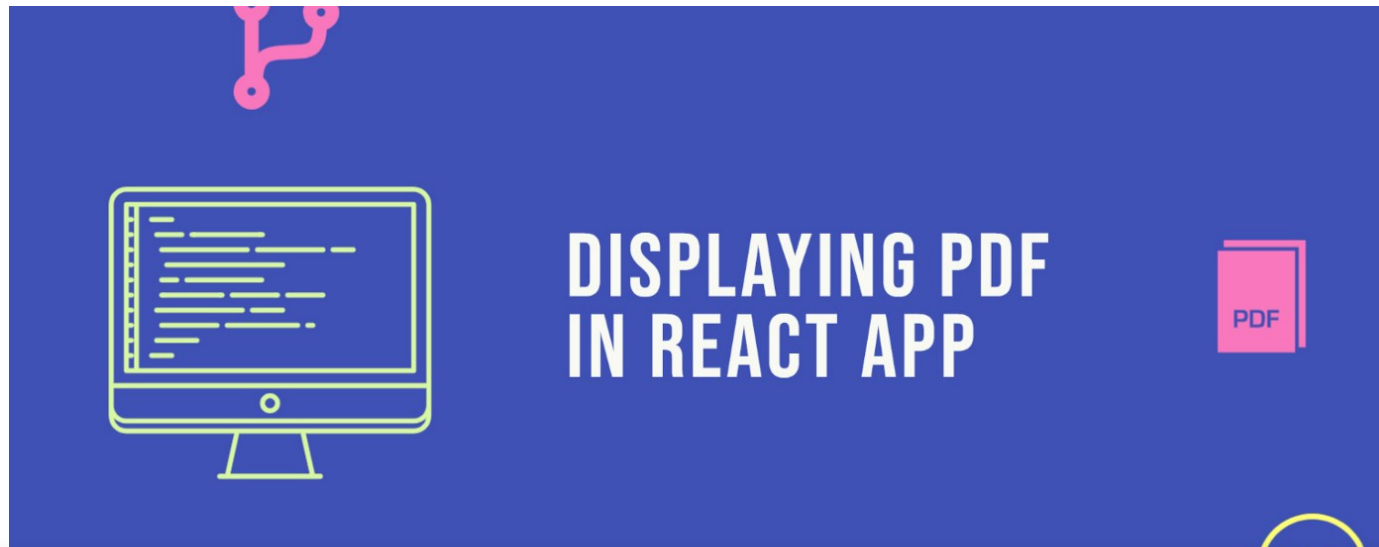
[Sign up for Medium and get an extra one](#)



Sagar Shrestha

[Follow](#)Jun 5, 2020 · 2 min read ★ · [Listen](#)

Displaying PDF in React app



[Sign In](#)[Get started](#)

I want to show you an easy and simple way to display `PDF` files using React in the browser. I will be using a library called react-pdf. This library is capable of rendering `PDF` files given an URL or a local file inside the project or base64 encoded version.

To get started, create a simple `React` application using Create React App. To do this you can run the following commands:

```
npx create-react-app display-pdf-app  
cd display-pdf-app  
npm start
```

After this, add the `react-pdf` library to the project using the following command:

```
npm install --save react-pdf
```



[Sign In](#)[Get started](#)

```
import React from 'react';
import { Document, Page } from 'react-pdf';

import samplePDF from './test.pdf';

export default function Test() {
  return (
    <Document file={samplePDF}>
      <Page pageNumber={1} />
    </Document>
  );
}
```

The final project structure will look like this:



[Sign In](#)[Get started](#)

```
public
index.html
pdf.worker.js
src
  components
    pdf
      all-pages.js
      single-page.js
  App.js
  index.js
  sample.pdf
  styles.css
package.json
```

Let's start with creating `single-page.js` and `all-pages.js` inside the `components` - `pdf` folder with the following contents:



[Sign In](#)[Get started](#)

```
1  import React, { useState } from "react";
2  import { Document, Page } from "react-pdf";
3
4  export default function SinglePage(props) {
5    const [numPages, setNumPages] = useState(null);
6    const [pageNumber, setPageNumber] = useState(1); //setting 1 to show first page
7
8    function onDocumentLoadSuccess({ numPages }) {
9      setNumPages(numPages);
10     setPageNumber(1);
11   }
12
13   function changePage(offset) {
14     setPageNumber(prevPageNumber => prevPageNumber + offset);
15   }
16
17   function previousPage() {
18     changePage(-1);
19   }
20
21   function nextPage() {
22     changePage(1);
23   }
24
25   const { pdf } = props;
26
27   return (
28     <>
```



[Sign In](#)[Get started](#)

```
34     <Page pageNumber={pageNumber} />
35   </Document>
36   <div>
37     <p>
38       Page {pageNumber || (numPages ? 1 : "--")} of {numPages || "--"}
39     </p>
40     <button type="button" disabled={pageNumber <= 1} onClick={previousPage}>
41       Previous
42     </button>
43     <button
44       type="button"
45       disabled={pageNumber >= numPages}
46       onClick={nextPage}
47     >
48       Next
49     </button>
50   </div>
51 </>
52 );
53 }
```

single-page.js hosted with ❤️ by GitHub

[view raw](#)

- `all-pages.js`



[Sign In](#)[Get started](#)

```
6
7  function onDocumentLoadSuccess({ numPages }) {
8    setNumPages(numPages);
9  }
10
11  const { pdf } = props;
12
13  return (
14    <Document
15      file={pdf}
16      options={{ workerSrc: "/pdf.worker.js" }}
17      onLoadSuccess={onDocumentLoadSuccess}
18    >
19      {Array.from(new Array(numPages), (el, index) => (
20        <Page key={`page_${index + 1}`} pageNumber={index + 1} />
21      ))}
22    </Document>
23  );
24 }
```



298



3

all-pages.js hosted with ❤️ by GitHub

[view raw](#)

Now navigate to `App.js` and replace the content with the following code:

```
1  import React from "react";
```



[Sign In](#)[Get started](#)

```
7  inside the app. Otherwise you can use the external link of the pdf file*/
8  import samplePDF from "./sample.pdf";
9
10 import "./styles.css";
11
12 export default function App() {
13   return (
14     <div className="App">
15       <h4>Single Page</h4>
16       <SinglePagePDFViewer pdf={samplePDF} />
17
18       <hr />
19
20       <h4>All Pages</h4>
21       <div className="all-page-container">
22         <AllPagesPDFViewer pdf={samplePDF} />
23       </div>
24
25       <hr />
26     </div>
27   );
28 }
```

App.js hosted with ❤ by GitHub

[view raw](#)

As you have noticed we have to create buttons to change the page of the

document in the PDFViewer component. This is because the PDFViewer does not



[Sign In](#)[Get started](#)

The full project code can be found here [github](#) and you can see the demo below:



[Sign In](#)[Get started](#)

If you get the following error in console:

```
Uncaught SyntaxError: Unexpected token <
```

```
http://project_server.dev/app_folder/d58d9dd3a5c5f2a8eafb.worker.js
```

```
...
```

then follow the steps below:

1. Copy `node_modules/pdfjs-dist/build/pdf.worker.js` to `public/ pdf.worker.js`
2. `options={{workerSrc: "pdf.worker.js"}}` Add this props inside the `Document` component from `react-pdf` . [It is already done in the above project.]

Reference



[Sign In](#)[Get started](#)

Level Up Coding

Thanks for being a part of our community! Level Up is transforming tech recruiting. [Find your perfect job](#) at the best companies.

Level Up — Transforming the Hiring Process

- 🔥 Enabling software engineers to find the perfect role that they love
- 🧠 Finding talent is the most painful part of...

jobs.levelup.dev

Sign up for Top Stories

By Level Up Coding

A monthly summary of the best stories shared in Level Up Coding [Take a look.](#)





[Sign In](#)

[Get started](#)

[About](#) [Help](#) [Terms](#) [Privacy](#)

