TRY CODING FUN

重拾编程的乐趣

🔍

在 React / NextJs 中使用 PDF.js

🗓 2022-03-25 | 🕮 2022-08-04 | 🗁 JavaScript

🅦 3.7k | ⏱ 3 分钟

需要在项目中用 pdf.js 预览 pdf 文件，并且将 pdf 文件转换为图片。

这儿以 React 为例

安装 `pdfjs-dist` ，并且复制 `pdf.worker.min.js` 到 `public/` 文件夹中，当然这是我的做法，如果你有更高阶的什么动态加载，打包到项目里，可以研究试试

```
1  npm install pdfjs-dist
2  cp ./node_modules/pdfjs-dist/build/pdf.worker.min.js ./public/
```

```
1   // pages/index.js
2   import React, { useEffect, useRef } from 'react';
3
4   export default function App() {
5         const canvasRef = useRef(null);
6
7         useEffect(() => {
8               (async function () {
9                     // We import this here so that it's only loaded during cli
10                    const pdfJS = await import('pdfjs-dist/build/pdf');
11                    pdfJS.GlobalWorkerOptions.workerSrc =
12                          window.location.origin + '/pdf.worker.min.js';
13                    const pdf = await pdfJS.getDocument('example.pdf').promise
14
15                    const page = await pdf.getPage(1);
16                    const viewport = page.getViewport({ scale: 1.5 });
17
18                    // Prepare canvas using PDF page dimensions.
19                    const canvas = canvasRef.current;
```

```
20                    const canvasContext = canvas.getContext('2d');
21                    canvas.height = viewport.height;
22                    canvas.width = viewport.width;
23
24                    // Render PDF page into canvas context.
25                    const renderContext = { canvasContext, viewport };
26                    page.render(renderContext);
27                })();
28        }, []);
29
30        return <canvas ref={canvasRef} style={{ height: '100vh' }} />;
31    }
```

Next.js 配置

- https://codesandbox.io/s/react-pdf-next-js-forked-0wyfr8

next.config.js

```
1   module.exports = {
2     future: {
3       webpack5: true
4     },
5     webpack: (config) => {
6       // load worker files as a urls with `file-loader`
7       config.module.rules.unshift({
8         test: /pdf\.worker\.(min\.)?js/,
9         use: [
10          {
11            loader: "file-loader",
12            options: {
13              name: "[contenthash].[ext]",
14              publicPath: "_next/static/worker",
15              outputPath: "static/worker"
16            }
17          }
18        ]
19      });
20
21      return config;
22    }
23  };
```

根目录新建一个 `pdf-worker.js`

```
1   if (process.env.NODE_ENV === "production") {
2     // use minified verion for production
3     module.exports = require("pdfjs-dist/build/pdf.worker.min.js");
4   } else {
5     module.exports = require("pdfjs-dist/build/pdf.worker.js");
6   }
```

页面文件 `pdf-viewer.js`

```
1    import { useState } from "react";
2    // import default react-pdf entry
3    import { Document, Page, pdfjs } from "react-pdf";
4    // import pdf worker as a url, see `next.config.js` and `pdf-worker.js`
5    import workerSrc from "../pdf-worker";
6
7    pdfjs.GlobalWorkerOptions.workerSrc = workerSrc;
8
9    export default function PDFViewer() {
10     const [file, setFile] = useState("./sample.pdf");
11     const [numPages, setNumPages] = useState(null);
12
13     function onFileChange(event) {
14       setFile(event.target.files[0]);
15     }
16
17     function onDocumentLoadSuccess({ numPages: nextNumPages }) {
18       setNumPages(nextNumPages);
19     }
20
21     return (
22       <div>
23         <div>
24           <label htmlFor="file">Load from file:</label>{" "}
25           <input onChange={onFileChange} type="file" />
26         </div>
27         <div>
28           <Document file={file} onLoadSuccess={onDocumentLoadSuccess}>
29             {Array.from({ length: numPages }, (_, index) => (
30               <Page
31                 key={`page_${index + 1}`}
32                 pageNumber={index + 1}
33                 renderAnnotationLayer={false}
```

```
34                  renderTextLayer={false}
35              />
36          ))}
37        </Document>
38      </div>
39    </div>
40  );
41 }
42
```

调用

```
1  import dynamic from "next/dynamic";
2
3  const PDFViewer = dynamic(() => import("../components/pdf-viewer"), {
4    ssr: false
5  });
6
7  export default function PDF() {
8    return <PDFViewer />;
9  }
```

其他方法

React-PDF

- https://projects.wojtekmaj.pl/react-pdf/
- https://www.npmjs.com/package/react-pdf

参考文章

- https://pspdfkit.com/blog/2021/how-to-build-a-reactjs-viewer-with-pdfjs/

相关文章

- Window Not Defined In NextJS 解决方法
- JS Vue React 是怎么处理路由的
- JS vue3 reactive 重新赋值的问题

- React 组件的父子传值

- 解决 Axios 额外的发起一次 option 请求

打赏

本文作者： Mt.r

本文链接： https://trycoding.fun/JavaScript/js-use-pdfjs-in-nextjs-reactjs/

版权声明： 本博客所有文章除特别声明外，均采用 ⓒ BY-NC-SA 许可协议。转载请注明出处!

\# JavaScript   \# Js   \# React   \# NextJs

‹ Js 转换 Canvas 为 Blob 或者 Image   JS 调用打印机的库，非常好用 ›