

報告書

PDF ファイルの canvas 表示について
豊橋技術科学大学 知能情報工学課程 3年 角 海音
令和2年 1月 22日

1. 目的

この報告書はPDF ファイルを Web ページの canvas に表示するまでの手順を記載したものである。PDF ファイルはサーバーにアップロードしたものを用いる。サーバーは node.js の簡易サーバ” http-server”を用いる。

2. 参考 URL

今回の試みでは次のサイトを参考に行った。以下に参考ページを記す。

- 『Mozilla の PDF.js を利用して JavaScript で PDF レンダリングをカスタマイズ - WPJ』
<https://www.webprofessional.jp/custom-pdf-rendering/undefined>

3. 実行方法

PDF.js から必要なファイル(pdf.js, pdf.worker.js)をダウンロードしサーバーにアップロードしたとする。以下に pdf ファイルを表示する html のコードを示す。これは参考ページを基に組み合わせたものであり、このまま実行すると何も表示されないページとなっている。修正する必要がある個所は黄色いマーカーが引かれている。また具体的にどのように修正するかは次章にて示す。

```
<!DOCTYPE html>
<html>
  <head>
    <title>PDF.js Learning</title>
  </head>
  <body>
    <script type="text/javascript" src="pdf.js"></script>
    <script type="text/javascript">
      PDFJS.workerSrc = "/path/to/pdf.worker.js";

      // URL of PDF document
      var url = "http://mozilla.github.io/pdf.js/examples/learning/helloworld.pdf";

      // Asynchronous download PDF
      PDFJS.getDocument(url)
        .then(function(pdf) {
          return pdf.getPage(1);
        })
        .then(function(page) {
          // Set scale (zoom) level
          var scale = 1.5;

          // Get viewport (dimensions)
          var viewport = page.getViewport(scale);

          // Get canvas#the-canvas
          var canvas = document.getElementById('the-canvas');

          // Fetch canvas' 2d context
          var context = canvas.getContext('2d');

          // Set dimensions to Canvas
          canvas.height = viewport.height;
          canvas.width = viewport.width;

          // Prepare object needed by render method
          var renderContext = {
```

```

        canvasContext: context,
        viewport: viewport
    };

    // Render PDF page
    page.render(renderContext);
});
</script>
<canvas id="the-canvas"></canvas>
</body>
</html>

```

4. 修正方針

前章にて示したソースコードにはいくつか手を加えないといけない箇所が大きく2つある。修正箇所には黄色のマーカーを引いてある。具体的にどのように修正すればよいか1か所ずつ説明する。

- **PDFJS** → pdfjsLib

これは参考記事が書かれた後に PDF.js の仕様に変更があったため必要となった修正箇所である。従って今後も PDF.js 内部のライブラリを用いたい場合、オブジェクト名はこのように記載する必要がある。

- <http://mozilla.github.io/pdf.js/examples/learning/helloworld.pdf> → (任意の pdf ファイルへのパス)

これはリンク切れのため修正する必要がある箇所である。参考ページでは Github に公開している pdf ファイルへのパスを記載しており外部のサーバーへのパスも有効であると考えるが、アクセス権限の関係から pdf ファイルへのパスはサーバーにアップロードしたものを用いることが無難であると考ええる。

5. 修正後のファイル

以下に修正後の html コードを示す。canvas に pdf ファイルを表示するためにはこのソースコードが最も簡潔である。

```

<!DOCTYPE html>
<html>
  <head>
    <title>PDF.js Learning</title>
  </head>
  <body>
    <script type="text/javascript" src="pdf.js"></script>
    <script type="text/javascript">
      pdfjsLib.workerSrc = "/path/to/pdf.worker.js";

      // URL of PDF document
      var url = "http://localhost:8080/compressed.tracemonkey-pldi-09.pdf";

      // Asynchronous download PDF
      pdfjsLib.getDocument(url)
        .then(function(pdf) {
          return pdf.getPage(1);
        })
        .then(function(page) {
          // Set scale (zoom) level
          var scale = 1.5;

          // Get viewport (dimensions)
          var viewport = page.getViewport(scale);

```

```
// Get canvas#the-canvas
var canvas = document.getElementById('the-canvas');

// Fetch canvas' 2d context
var context = canvas.getContext('2d');

// Set dimensions to Canvas
canvas.height = viewport.height;
canvas.width = viewport.width;

// Prepare object needed by render method
var renderContext = {
  canvasContext: context,
  viewport: viewport
};

// Render PDF page
page.render(renderContext);
});
</script>
<canvas id="the-canvas"></canvas>
</body>
</html>
```