# Generating PDF on the backend

Status	DONE			
Owner	@Sławomir Pawluk			
Contributors	@Colm Nolan			
Approved	@Piotr Litwinski			
Decision	Decision is to go with IronPdf and monthly subscription licensing model			
Recording	TDA to discuss PDF Viewer-20240509_103319-Meeting Recording.mp4			
On this page	Problem statement Research insights In Solution hypothesis Follow up Source files			

## ? Problem statement

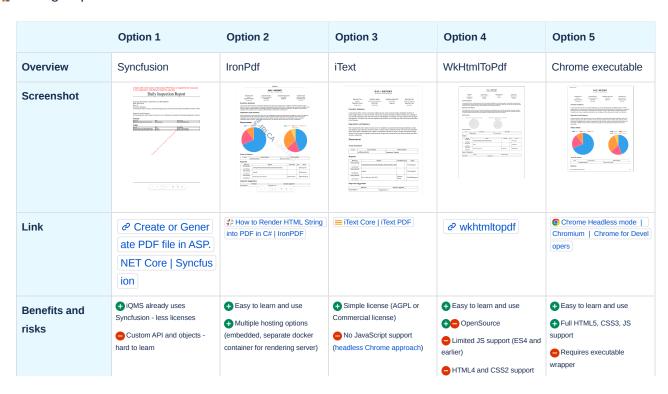
Research insights

Solution hypothesis

Inspection reports should be generated by API, to hide business logic. Reports should be scalable and easy to maintain.

 We assume that a single customer will generate 500 PDF documents a month

# Design options

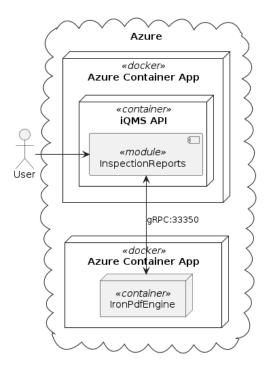


Crit eria	Page header/foo ter support	Supported with page numbering	Full HTML5, CSS3, JS support Price Complex license terms  Supported with page numbering	□ Limited HTML and CSS support □ Requires pre-rendering charts □ Limited fonts support □ Very complex digital signature process  Supported with page numbering	Requires executable wrapper  No digital signature support  Supported with page numbering	Poor documentation No digital signature  Not supported
	Watermark ing	Not tested (due to charts generation not supported)	Supported	Supported	Supported (CSS and HTML background)	Supported (CSS and HTML background)
	Charts generation	Not supported (therefore not fit for purpose)	Supported by JavaScript evaluation	Supported by JavaScript evaluation using headless Chrome (Selenium driver)	Supported by JavaScript evaluation using legacy version of JS (ES4 and older)	Supported by JavaScript evaluation
	Digital signature	Not tested (due to charts generation not supported)	Supported (.p12 and .pfx)	Supported	Not supported	Not supported
	Price	\$695/month/10 devs	\$2999/10 devs - IronPdf \$1999 - SaaS & OEM redistribution *\$999/year (1 year free) or \$1999/5y - upgrades  Total: \$4998-\$6997  Unlimited Subscription \$500/month or more. • 10,000 API calls included • \$0.02 per call after • Redistribution included • All suite products • Updates included • Priority support • No contract	Volume-based (annual subscription)     OEM Distribution license (Custom)  Price is not disclosed	\$0	\$0

## **Syncfusion PoC**

#### **IronPdf**

Architecture:



### Code example:

```
var renderer = new ChromePdfRenderer();
2
3 var pdf = await renderer.RenderHtmlAsPdfAsync(html);
4
5 pdf.AddHtmlHeaders(new HtmlHeaderFooter()
6 {
7
       MaxHeight = 15,
8
       HtmlFragment = "<center>Header</center>",
9
       DrawDividerLine = true,
10 });
11
12 pdf.AddHtmlFooters(new HtmlHeaderFooter()
13 {
14
       MaxHeight = 15,
15
       HtmlFragment = "<center><i>{page} of {total-pages}<i></center>",
16
       DrawDividerLine = true,
17
18 });
19
20 var watermarkHtml = @"<img src='https://www.irlca.com/wp-content/uploads/2020/04/Irlca-logo-v3@2x-1.png'/>";
21 pdf.ApplyWatermark(watermarkHtml, rotation: 45);
22
23 var cert = new X509Certificate2(Path.Combine(".", "data", "cert.pfx"), "pass",
   X509KeyStorageFlags.Exportable);
24 var signature = new PdfSignature(cert);
25 signature.TimestampHashAlgorithm = TimestampHashAlgorithms.SHA256;
26 signature.TimeStampUrl = "http://timestamp.digicert.com";
27
28 pdf.Sign(signature);
29 pdf.SaveAs("result.pdf");
```

Watermarks: https://ironpdf.com/how-to/custom-watermark/

Signing: https://ironpdf.com/how-to/signing/

#### **iText**

Code example:

```
using var dest = File.Open("result.pdf", FileMode.Create);
var properties = new ConverterProperties();
HtmlConverter.ConvertToPdf(html, dest, properties);
```

Watermarks: Watermark examples

Signing: https://kb.itextpdf.com/itext/digital-signatures-chapter-4

### WkHtmlToPdf

```
1 .\wkhtmltopdf.exe -s A4 --enable-javascript --no-stop-slow-scripts --debug-javascript --javascript-delay 1000 sample.html sample.pdf
```

### **Headless Chrome**

"C:\Program Files\Google\Chrome\Application\chrome.exe" --headless --print-to-pdf=c:\dest\output.pdf
c:\src\sample.html

# ✓ Follow up

Decision	Status	Next steps
	DECIDED / IN REVIEW / OTHER	

## Source files

Type /link to add links to design files.