PENTHRA

DOCUMENTACIÓN TÉCNICA DEL FABRICANTE

PENTHRA PDF SDK 1.0 — Guía técnica de instalación y uso

Java y .NET/C# — PENTHRA TECH E.I.R.L.

Código ING-2026-003
Versión 1.0 · 19 de octubre de 2026
Producto PENTHRA PDF SDK 1.0.0
Clasificación Uso del cliente

01 Introducción

Esta guía es la documentación técnica del fabricante para instalar, activar y usar el **PENTHRA PDF SDK 1.0.0** en aplicaciones Java y .NET. Está dirigida a desarrolladores y administradores de la Entidad. La referencia completa de clases y métodos se entrega como Javadoc (Java) y documentación XML (.NET), y los ejemplos de código en la carpeta samples.

Elemento	Descripción
Librería Java	penthra-pdf-sdk-1.0.0.jar + carpeta lib/ (dependencias). Maven: tech.penthra:penthra-pdf-sdk:1.0.0
SDK .NET	Paquete NuGet Penthra.Pdf 1.0.0 (netstandard2.0 y net8.0), incluye el motor PENTHRA
Línea de comandos	java -jar penthra-pdf-sdk-1.0.0.jar: operaciones, flujos por lotes y servicio REST
Servicio REST	API HTTP local (127.0.0.1) con token, para integración con cualquier plataforma
Licencia	Archivo .lic firmado digitalmente, emitido a nombre de la Entidad

02 Requisitos

Componente	Requisito
Sistema operativo	Windows Server 2016 o superior, Windows 10/11, Linux x64 (RHEL, Ubuntu, Debian), macOS para desarrollo
Java	Java Runtime 11 o superior (recomendado 17 o 21 LTS, p. ej. Eclipse Temurin)
.NET	.NET Framework 4.6.2 o superior, .NET 6 o .NET 8 (el SDK .NET usa el mismo motor Java)
Memoria	512 MB libres como mínimo por proceso; 1 a 2 GB para OCR y documentos grandes
OCR	Tesseract 5 con modelos spa y eng (Windows: instalador UB-Mannheim; Linux: paquetes tesseract-ocr y tesseract-ocr-spa)
Red	No requiere internet. Solo la firma con sello de tiempo necesita acceso a la TSA configurada

03 Instalación

Java

Opción 1, repositorio Maven interno de la Entidad: publicar el jar y su pom, y declarar la dependencia:

```
<dependency>
  <groupId>tech.penthra</groupId>
  <artifactId>penthra-pdf-sdk</artifactId>
  <version>1.0.0</version>
</dependency>
```

Opción 2, sin Maven: copiar penthra-pdf-sdk-1.0.0.jar y la carpeta lib/ al classpath de la aplicación (el manifiesto del jar ya referencia lib/).

.NET

```
dotnet add package Penthra.Pdf --version 1.0.0 --source <carpeta-o-feed-interno>
```

Al compilar, el motor se copia a la carpeta penthra-engine/ junto a la aplicación. En IIS, verifique que la identidad del grupo de aplicaciones pueda ejecutar java (o configure EngineOptions.JavaPath).

OCR

Instale Tesseract 5 y defina la variable TESSDATA_PREFIX con la carpeta tessdata (o use Ocr.Options.tessdata() / EngineOptions.TessdataPath). En Windows agregue la carpeta de Tesseract al PATH para que se cargue libtesseract.

04 Licencia y activación

El SDK busca la licencia en este orden: propiedad de sistema penthra.pdf.license, variable de entorno PENTHRA_PDF_LICENSE (ruta o contenido) y archivo penthra-pdf.lic en el directorio de trabajo. También puede cargarse por código con PenthraPdf.license(ruta). Sin licencia válida el SDK funciona en modo evaluación: los documentos llevan la marca "EVALUACIÓN".

Concepto	Detalle
Contenido	Titular, RUC, correo de registro, módulos, vigencia (inicio y fin), volumen anual, ambientes y condiciones de soporte
Seguridad	Firma digital ECDSA P-256: cualquier alteración invalida la licencia
Ambientes	PENTHRA_PDF_ENV o -Dpenthra.pdf.environment: production (consume volumen); development, testing, qa, certification y staging (no consumen)
Evento	Cada documento generado o guardado por el SDK en producción (guardado, firma, conversión, reparación)
Consumo	java -jar penthra-pdf-sdk-1.0.0.jar license, GET /api/v1/license, o LicenseManager.get().status()
Contador	Archivo en ~/.penthra-pdf o PENTHRA_PDF_USAGE_DIR; seguro entre hilos y procesos del mismo servidor
Exceso de volumen	Modo SOFT (por defecto): se registra una advertencia y la operación continúa, para no interrumpir el servicio
Año de licencia	Se reinicia en cada aniversario de la fecha de inicio

05 Primeros pasos en Java

```
PenthraPdf.license(Paths.get("penthra-pdf.lic"));

PdfDoc doc = new DocumentBuilder()
    .title("Constancia").lang("es-PE")
    .pdfA(HtmlConverter.PdfA.PDFA_2A).pdfUa(true)
```

```
    .header("Ministerio de Educación").footerPageNumbers()
    .heading(1, "Constancia de estudios")
    .paragraph("Se deja constancia que ...")
    .table(List.of("Área", "Nota"), List.of(List.of("Matemática", "18")))
    .qr("https://www.minedu.gob.pe/verificar/ABC123", 80, "QR de verificación")
    .build();
doc.save(Paths.get("constancia.pdf"));

try (PdfDoc d = PenthraPdf.open(Paths.get("constancia.pdf"))) {
    Watermarks.text(d, "COPIA", new Watermarks.Options().opacity(0.2f));
    Encryption.encrypt(d, "", "clave", Encryption.Permissions.readOnly(),
Encryption.Algorithm.AES_256);
    d.save(Paths.get("protegido.pdf"));
}
```

PdfDoc no es seguro entre hilos: use una instancia por hilo o solicitud. Cierre los documentos (try-with-resources) para liberar memoria.

06 Primeros pasos en .NET

```
using (var pdf = PdfEngine.Start(new EngineOptions { LicensePath = "penthra-pdf.lic" }))
{
    byte[] doc = pdf.HtmlToPdf(html, new HtmlOptions { PdfUa = true, PdfA = PdfAConformance.PdfA2A });
    doc = pdf.Watermark(doc, "COPIA");
    doc = pdf.Sign(doc, File.ReadAllBytes("firma.p12"), "clave",
        new SignOptions { Reason = "Conformidad", Visible = true, X = 330, Y = 60 });
    File.WriteAllBytes("constancia.pdf", doc);
}
```

PdfEngine es seguro entre hilos y conviene crearlo una sola vez por aplicación (por ejemplo, como singleton en el contenedor de dependencias). Para un despliegue centralizado use PdfEngine.Connect(url, token) contra el servicio REST.

07 Referencia por módulo

Necesidad	Java (tech.penthra.pdf)	.NET (PdfEngine)
Diseñar documentos	document.DocumentBuilder	HtmlToPdf
HTML/CSS y SVG a PDF	convert.HtmlConverter, convert.SvgConverter	HtmlToPdf, SvgToPdf, DrawSvg
Combinar, dividir, páginas	manipulate.Pages, pages.PageInfo	Merge, Split, SplitByRanges, ExtractPages, DeletePages, Rotate, Reorder, PageCount, FindTextPages
Marcas, encabezados, códigos	manipulate.Watermarks, barcode.Barcodes	Watermark, WatermarkImage, HeaderFooter, AddBarcode, ReadBarcodes
Formularios	forms.AcroForms, xfa.Xfa	FormFields, FillForm, FlattenForm, ExportXfdf, ImportXfdf, XfaInfo, XfaDataXml, XfaToAcroForm, XfaToPdf
Anotaciones	annotations.Annotations	AddNote, AddHighlight, FlattenAnnotations

Necesidad	Java (tech.penthra.pdf)	.NET (PdfEngine)
Extracción	extract.TextExtraction, ImageExtraction, TableExtraction, StructuredExtraction	ExtractText, ExtractTables, ExtractJson
Etiquetado y accesibilidad	tagged.TaggedPdf, pdfua.PdfUa	Structure, AccessibilityReport, FixPdfUaDocumentLevel
Preservación	pdfa.PdfA, validate.Validator	ConvertToPdfA, Validate
Seguridad	security.Encryption, sign.Signer, sign.SignatureValidator, redact.Redactor	Encrypt, Decrypt, Sign, VerifySignatures, Redact
Escrituras complejas	text.ComplexText	HtmlToPdf con fuentes registradas
Inspección, OCR, optimización	inspect.Inspector, ocr.Ocr, optimize.Optimizer	Inspect, Ocr, Optimize
Bajo nivel	lowlevel.LowLevel, PdfDoc.pd()	Run(operación, ...) y flujos
Automatización	workflow.Workflow	RunWorkflow(new Workflow().Step(...))

Firma digital con certificado del token o HSM (PKCS#11)

```
KeyStore ks = KeyStore.getInstance("PKCS11", Security.getProvider("SunPKCS11").configure("token.cfg"));
ks.load(null, pin);
Signer.Credential cred = Signer.Credential.fromKeyStore(ks, pin, null);
byte[] firmado = Signer.sign(doc, cred, new Signer.Options().reason("Aprobado")
    .visible(1, new PDRectangle(330, 60, 220, 60)).timestampUrl("https://tsa.ejemplo.gob.pe"));
```

Anonimización con verificación

```
Redactor.Report r = Redactor.redact(doc, new Redactor.Options()
    .rule(Redactor.Rule.dniPeru()).rule(Redactor.Rule.email())
    .rule(Redactor.Rule.regex("Expediente", "EXP-\\d{4}-\\d+"))
    .ocrImages(true, "spa").overlayText("REDACTADO"));
if (!r.verified()) { /* revisar */ }
```

08 Línea de comandos y API REST

```
java -jar penthra-pdf-sdk-1.0.0.jar ops
java -jar penthra-pdf-sdk-1.0.0.jar op pdfa -i entrada.pdf -p level=\"PDFA_2B\" -o salida.pdf
java -jar penthra-pdf-sdk-1.0.0.jar workflow flujo.json -I anexo=anexo.pdf -o resultado.pdf
java -jar penthra-pdf-sdk-1.0.0.jar serve --host 127.0.0.1 --port 8765 --token SECRETO
```

Método y ruta	Descripción	Cuerpo
GET /api/v1/health	Estado y versión	—
GET /api/v1/license	Suscripción y consumo del año	—
GET /api/v1/operations	Operaciones disponibles	—

Método y ruta	Descripción	Cuerpo
POST /api/v1/ops/{operación}	Ejecuta una operación	JSON con "input" (Base64) y parámetros
POST /api/v1/workflow	Ejecuta un flujo	JSON con "steps" e "inputs"
POST /api/v1/convert/html	HTML a PDF (?pdfua=true&pdfa=PDFA_2B)	HTML; responde application/pdf

Todas las rutas exigen el encabezado Authorization: Bearer <token> cuando el servicio se inicia con token. El servicio escucha por defecto solo en 127.0.0.1.

09 Flujos de trabajo

Un flujo es un JSON con una lista de pasos que se ejecutan sobre el documento actual; los resultados (reportes, texto, datos) se devuelven en "results". Operaciones disponibles: open, html-to-pdf, svg-to-pdf, draw-svg, merge, split, extract-pages, delete-pages, rotate, reorder, watermark, header-footer, barcode, read-barcodes, form-fields, fill-form, flatten-form, export-xfdf, import-xfdf, xfa-info, xfa-data, xfa-to-acroform, xfa-to-pdf, annotate-note, annotate-highlight, flatten-annotations, redact, pdfa, validate, pdfua-fix, accessibility, structure, ocr, optimize, encrypt, decrypt, security-info, sign, verify-signatures, inspect, extract-text, extract-tables, extract-json, page-count, find-text-pages, metadata y save. La Entidad puede registrar pasos propios con Workflow.register().

```
{"steps": [
  {"op": "html-to-pdf", "html": "<h1>Constancia</h1>", "pdfua": true},
  {"op": "watermark", "text": "COPIA CONTROLADA", "opacity": 0.2},
  {"op": "redact", "patterns": ["DNI", "EMAIL"]},
  {"op": "pdfa", "level": "PDFA_2B"},
  {"op": "validate", "profile": "PDFA_2B"}
], "outputPath": "constancia.pdf"}
```

10 Seguridad y buenas prácticas

- Ejecute el SDK con java.awt.headless=true en servidores (el SDK .NET lo hace automáticamente).
- Proteja el archivo de licencia, los certificados de firma y el token del servicio REST con permisos de sistema operativo; no los incluya en repositorios de código.
- Para firmar con certificados de funcionarios use tokens o HSM vía PKCS#11; la clave privada no sale del dispositivo.
- El servicio REST no debe exponerse a internet; si se centraliza, publíquelo detrás de un proxy con TLS y control de acceso.
- Valide los PDF/A y PDF/UA con Validator antes de archivarlos o publicarlos.
- Al anonimizar, conserve el reporte de verificación (remainingMatches = 0) como sustento.

11 Soporte, actualizaciones y garantía

Concepto	Condición
Horario	Lunes a viernes de 08:15 a 17:15 (hora de Perú)
Canales	Correo soporte@penthra.tech, teléfono [número de soporte] y plataforma de incidencias [URL]
Tiempo de respuesta	Máximo 24 horas hábiles desde el registro de la incidencia
Actualizaciones	Nuevas versiones, correcciones y parches incluidos durante los 36 meses; versionado semántico (1.x compatible)
Garantía	Reemplazo de la suscripción (nuevo archivo de licencia) en un máximo de 72 horas
Información para soporte	Versión (java -jar ... --version), estado de licencia (license), reporte de Inspector y archivo de muestra si es posible

12 Solución de problemas

Síntoma	Causa y solución
Los PDF salen con la marca "EVALUACIÓN"	No se cargó la licencia o venció: verifique la ruta y el resultado de "license"
"No se encontró la carpeta tessdata"	Instale Tesseract y defina TESSDATA_PREFIX
UnsatisfiedLinkError: tesseract	Falta la biblioteca nativa: agregue Tesseract al PATH (Windows) o instale libtesseract (Linux)
Validación PDF/A falla por fuentes	El documento usa fuentes no incrustadas: convierta con FontPolicy.RASTERIZE_PAGES o regenere el original
El motor .NET no inicia	Java no está instalado o no es accesible: configure EngineOptions.JavaPath o PENTHRA_PDF_JAVA
Firma "no confiable"	La cadena del certificado no llega a una raíz de confianza: indique las raíces de la ECEP en la validación
Falta un carácter en el PDF	La fuente no cubre ese alfabeto: registre una fuente adecuada (p. ej. Noto) en DocumentBuilder.font()