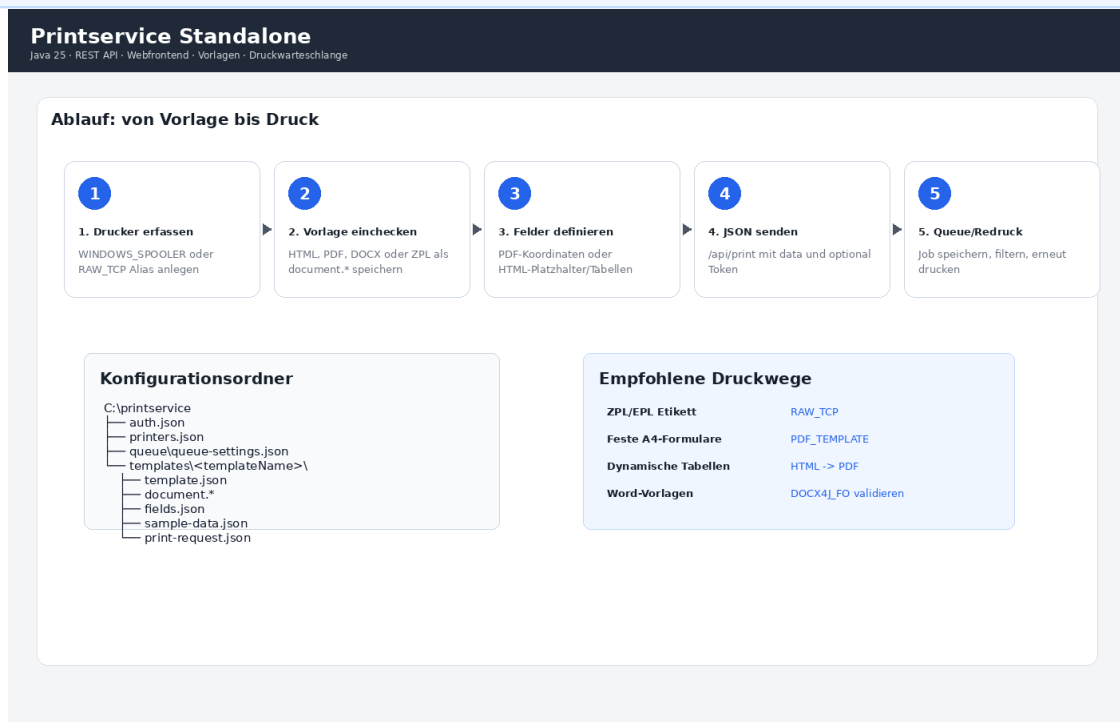


Printservice Standalone Java 25 v2.5

API- und Vorlagendokumentation mit Beispielen

Eigenständige Java-Anwendung mit Webfrontend, REST-API, Drucker-Aliasverwaltung, optionaler Authentifizierung, Druckwarteschlange/Redruck, HTML/PDF/DOCX/ZPL-Vorlagen und Barcode-/QR-Code-Unterstützung.



Überblick: Vom Drucker-Alias über die Vorlage bis zum Druckjob.

Stand: Version v2.5, erstellt am 10.06.2026. Diese Dokumentation beschreibt die Standalone-Variante ohne GlassFish/Payara.

Inhalt

1. Überblick und Architektur
2. Installation und Windows-Service
3. Konfigurations- und Speicherstruktur
4. Bedienung im Webfrontend
5. Vorlagen erstellen: HTML, PDF_TEMPLATE, DOCX und ZPL
6. Authentifizierung und Benutzer/Token
7. Drucker und Aliasverwaltung
8. Drucken, Vorschau und Render-Modi
9. Druckwarteschlange, Redruck und Datenschutz
10. Vollständige REST-API-Referenz
11. Beispiel-Payloads und cURL-Aufrufe
12. Barcode-/QR-Code-Arten und Platzhalter
13. Fehlerbehebung und Produktionshinweise

1. Überblick und Architektur

Der Printservice ist ab v2.x eine eigenständige Java-25-Applikation. Er benötigt keinen Application Server. Der integrierte HTTP-Server liefert das Webfrontend aus und stellt unter /api die REST-Schnittstellen bereit.

Komponente	Aufgabe
Webfrontend	Drucker anlegen, Vorlagen einchecken, JSON testen, Queue/Redruck bedienen, Auth verwalten.
REST-API	Maschinen, HMI, MES oder externe Anwendungen senden Druckjobs per HTTP/JSON.
TemplateService	Speichert Vorlagen als Ordnerstruktur unter templates/.
PrintJobService	Rendert RAW, HTML->PDF, PDF_TEMPLATE oder DOCX und sendet an Drucker.
PrinterRegistry	Verwaltet Aliase für WINDOWS_SPOOLER und RAW_TCP.
PrintQueueService	Persistente Historie für Redruck, Filterung und Datenschutz-Bereinigung.
AuthService	Optionale Benutzer-/Tokenverwaltung mit Scopes PRINT, PREVIEW, ADMIN, ALL.

Die Renderpfade sind bewusst getrennt: ZPL/EPL wird möglichst direkt als RAW_TCP gesendet, PDF/HTML/PDF_TEMPLATE nutzt PDF-Rendering und Windows-Spooler/CUPS, DOCX kann als DOCX erzeugt oder über DOCX4J_FO nach PDF konvertiert werden.

2. Installation und Windows-Service

Build

```
cd printservice-standalone-java25
mvn clean package

# Ergebnis
target/printservice-standalone.jar
```

Start als Konsole

```
# Windows
java -Dprintservice.config.dir=C:\printservice ^
-jar target\printservice-standalone.jar ^
--host=0.0.0.0 ^
--port=8080

# Linux
java -Dprintservice.config.dir=/opt/printservice -jar target/printservice-standalone.jar
--host=0.0.0.0 --port=8080
```

Windows-Service über WinSW

Die Anwendung ist für WinSW vorbereitet. Die Skripte befinden sich im Projekt unter scripts/, die Service-XML unter service/printservice.xml.

```
mvn clean package
.\scripts\install-service.ps1 -InstallDir C:\printservice-app -ConfigDir C:\printservice
-Port 8080

cd C:\printservice-app
.\printservice.exe install
.\printservice.exe start
.\printservice.exe status
```

Für Produktion sollte der Konfigurationsordner explizit gesetzt werden. Ohne Angabe wird ein Benutzerprofilpfad verwendet, was im Dienstbetrieb schwer zu administrieren ist.

3. Konfigurations- und Speicherstruktur

```
C:\printservice
├─ auth.json
```

```

| printers.json
| queue
|   | queue-settings.json
|   | print-jobs.json
|   | templates
|   |   | <templateName>
|   |   | template.json
|   |   | document.*
|   |   | fields.json
|   |   | sample-data.json
|   |   | print-request.json
|   | spool
|   | archive
|   | preview
|   | logs

```

Datei/Ordner	Bedeutung
auth.json	Optionale Authentifizierung, User und Token-Hashes.
printers.json	Drucker-Aliase für WINDOWS_SPOOLER und RAW_TCP.
templates//template.json	Metadaten der Vorlage.
templates//document.*	Originaldokument: HTML, PDF, DOCX, ZPL, EPL, TXT.
templates//fields.json	Koordinatenfelder für PDF_TEMPLATE.
templates//sample-data.json	Beispiel-JSON zur Vorschau.
templates//print-request.json	Vollständiger Beispiel-Druckrequest.
queue/queue-settings.json	Queue/Datenschutz-Einstellungen.
queue/print-jobs.json	Persistente Druckhistorie für Redruck.

4. Bedienung im Webfrontend

Benutzer und Token anlegen

[Bild fehlt: /mnt/data/printservice_doc_assets/mockups/01_auth_user_token.png]

Empfohlener Ablauf: erst einen Admin-User mit Scope ALL anlegen, Token sichern, danach Auth aktivieren. Für Maschinen/Stationen separate User mit Scope PRINT oder PRINT+PREVIEW anlegen.

Windows-/Server-Drucker-Assistent

[Bild fehlt: /mnt/data/printservice_doc_assets/mockups/02_printer_assistant.png]

Drucker-Alias vergeben

[Bild fehlt: /mnt/data/printservice_doc_assets/mockups/03_printer_alias.png]

Vorlagenanlage und Drucktest

Printservice Standalone

Schritt 1: Dokument einchecken oder HTML online erstellen

3. Dokumentenvorlage einchecken oder online erstellen

Dokument einchecken

HTML-Beispiel

HTML-Tabellenbeispiel

ZPL-Beispiel

PDF-Feldbeispiel

Vorlagenname / Dokumentname

lieferschein_html_table

Beschreibung

Lieferschein als HTML-Tabelle

Content-Type

text/html

Vorlageninhalt / Quelltext

```
<!doctype html>
<html>
<body>
<h1>Lieferschein {{deliveryNumber}}</h1>
<table>
<thead><tr><th>Pos.</th><th>Artikel</th><th>Menge</th><th>Barcode</th></tr></thead>
<tbody>
{{#each:items}}
<tr>
<td>{{text:pos}}</td>
<td>{{text:partNumber}}</td>
<td>{{text:quantity}}</td>
<td></td>
</tr>
</each>
</tbody>
</table>
</body>
</html>
```

Vorlage speichern

Vorlagen laden

Einfacher HTML-Designer

Lieferschein LS-2026-0001

Pos.	Artikel	Menge	Barcode
1	A123456	10	
2	B987654	5	
3	C555111	20	

ML aus Quelltext uebernehmen

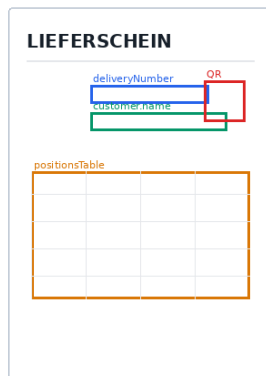
Webfrontend: HTML-Vorlage mit Online-Editor und Tabellen-Wiederholungsblock.

Printservice Standalone

Schritt 2: PDF als Vorlage plus Koordinatenfelder

PDF als Vorlage mit Koordinatenfeldern

document.pdf - statisches Layout



fields.json

```
{
  "coordinateUnit": "mm",
  "coordinateOrigin": "TOP_LEFT",
  "pdfFields": [
    { "name": "deliveryNumber", "type": "TEXT",
      "value": "{{deliveryNumber}}", "page": 1,
      "x": 38, "y": 35, "fontSize": 10 },
    { "name": "serialQr", "type": "QR_CODE",
      "value": "{{deliveryNumber}}",
      "x": 162, "y": 25, "width": 28, "height": 28 },
    { "name": "positionsTable", "type": "TABLE",
      "source": "items", "x": 18, "y": 78,
      "width": 174, "height": 125, "overflow": "CUT",
      "columns": [
        { "title": "Pos.", "field": "pos", "width": 12 },
        { "title": "Artikel", "field": "partNumber", "width": 29 },
        { "title": "Barcode", "type": "BARCODE",
          "barcodeType": "CODE_128", "field": "serialNumber" }
      ]
    }
  ]
}
```

PDF hochladen

PDF-Tabellenbeispiel

Speichern

Webfrontend: PDF_TEMPLATE mit fields.json und Koordinatenfeldern.

Printservice Standalone

Schritt 3: Word-Dokument als Druckvorlage verwenden

DOCX als Vorlage - Platzhalter in Word/LibreOffice setzen

document.docx

LIEFERSCHEIN

Lieferschein: {{deliveryNumber}}

Datum: {{deliveryDate}}

Kunde: {{customer.name}}

Artikel: {{partNumber}}

Menge: {{quantity}}

Barcode als eigener Absatz:

{{barcode:CODE_128:serialNumber:360x90}}

QR-Code als eigener Absatz:

{{qr:serialNumber}}

Regeln fuer DOCX-Vorlagen

- Normale Textfelder: {{serialNumber}}, {{customer.name}}
- Barcode/QR-Code jeweils allein in einem Absatz platzieren.
- renderMode=DOCX erzeugt ein DOCX ohne Konvertierung.
- docxConverter=DOCX4J_FO wandelt DOCX serverseitig nach PDF.
- Jede DOCX-Vorlage vor Produktivdruck validieren.

```
POST /api/docx/validate?converter=DOCX4J_FO
POST /api/print
```

```
{
  "printerAlias": "OFFICE_A4",
  "templateName": "lieferschein_docx",
  "renderMode": "DOCX",
  "docxConverter": "DOCX4J_FO",
  "data": { "serialNumber": "SNR000001" }
}
```

Webfrontend/Workflow: DOCX in Word/LibreOffice pflegen und als Vorlage einchecken.

Printservice Standalone

Redruck, Filterung und automatische Bereinigung

Druckwarteschlange / Historie / Redruck

Queue-Einstellungen

enabled true
storeRequestsForReprint true
autoCleanupEnabled true
retentionDays 30
Einstellungen speichern Queue komplett leer

Wichtige REST-Aufrufe

```
GET /api/queue?limit=100
GET /api/queue?full=true&user=line01
POST /api/queue/{jobid}/reprint
DELETE /api/queue
POST /api/queue/cleanup?olderThanDays=30
PUT /api/queue/settings
```

Queue-Liste mit User-Filter und Redruck

Zeit	Status	User	Drucker	Vorlage	Kopien	Job-ID	Aktion
2026-06-10 14:30	SUBMITTED	line01	OFFICE_A4	lieferschein_pdf_simple	1	JOB-0001	Redruck
2026-06-10 14:42	FAILED	line02	LABEL_LINE_01	serial_label_zpl	1	JOB-0002	Details
2026-06-10 14:55	SUBMITTED	line01	OFFICE_A4	lieferschein_html_table	2	JOB-0003	Redruck

Webfrontend: Queue/Redruck mit Filterung, Aufbewahrung und Datenschutzoptionen.

5. Vorlagen erstellen

5.1 HTML-Vorlage mit Tabelle

HTML-Vorlagen sind flexibel und eignen sich besonders für dynamische Tabellen. Der Printservice rendert Platzhalter, erzeugt Barcode-/QR-Code-Bilder als Data-URI, normalisiert HTML zu XHTML und

erzeugt daraus ein PDF.

```
<!doctype html>
<html>
<head>
<meta charset="utf-8">
<title>Lieferschein HTML Tabelle</title>
<style>
@page { size: A4; margin: 14mm; }
body { font-family: Arial, sans-serif; font-size: 10pt; color: #111; }
h1 { font-size: 18pt; margin: 0 0 8mm 0; }
.header { display: flex; justify-content: space-between; align-items: flex-start; }
.box { border: 1px solid #999; padding: 3mm; margin-bottom: 5mm; }
table { width: 100%; border-collapse: collapse; table-layout: fixed; }
th, td { border: 1px solid #777; padding: 2mm; vertical-align: top; }
th { background: #eeeeee; font-weight: bold; }
.right { text-align: right; }
.center { text-align: center; }
.barcode { width: 42mm; height: 12mm; object-fit: contain; }
.qr { width: 28mm; height: 28mm; }
</style>
</head>
<body>
<div class="header">
<div>
<h1>Lieferschein</h1>
<div><b>Nr.:</b> {{text:deliveryNumber}}</div>
<div><b>Datum:</b> {{text:deliveryDate}}</div>
</div>

</div>

<div class="box"><b>Kunde:</b> {{text:customer.name}}</div>

<table>
<thead>
<tr>
<th style="width:10mm">Pos.</th>
<th style="width:32mm">Artikel</th>
<th>Bezeichnung</th>
<th style="width:18mm">Menge</th>
<th style="width:38mm">Seriennummer</th>
<th style="width:45mm">Barcode</th>
</tr>
</thead>
<tbody>
{{#each:items}}
<tr>
<td class="center">{{text:pos}}</td>
<td>{{text:partNumber}}</td>
<td>{{text:description}}</td>
<td class="right">{{text:quantity}}</td>
<td>{{text:serialNumber}}</td>
<td class="center"></td>
</tr>
{{/each}}
</tbody>
</table>
</body>
</html>
```

Lieferschein

Nr.: LS-2026-0001
Datum: 08.06.2026
Kunde: Muster GmbH

Pos.	Artikel	Bezeichnung	Menge	Seriennummer	Barcode
1	A123456	Gehäuse Oberteil	10	SNR000001	
2	B987654	Elektronikmodul	5	SNR000002	
3	C555111	Dichtung	20	SNR000003	

Gerendertes Beispiel: HTML-Lieferschein mit dynamischer Tabelle und Barcode je Zeile.

5.2 PDF_TEMPLATE: PDF als feste Druckvorlage

Bei PDF_TEMPLATE wird ein fertiges PDF als Hintergrund verwendet. Text, Barcode, QR-Code und Tabellen werden anhand von Koordinaten in das PDF gestempelt. Das ist schnell und stabil für feste Formulare.

```
{
  "name": "lieferschein_pdf_simple",
  "documentName": "Lieferschein PDF einfach",
  "templateType": "PDF_TEMPLATE",
  "contentType": "application/pdf",
  "sourceFileName": "lieferschein_pdf_simple_template.pdf",
  "documentFile": "document.pdf",
  "coordinateUnit": "mm",
  "coordinateOrigin": "TOP_LEFT"
}

{
  "coordinateUnit": "mm",
  "coordinateOrigin": "TOP_LEFT",
  "pdfFields": [
    {
      "name": "deliveryNumber",
      "type": "TEXT",
      "value": "{{deliveryNumber}}",
      "page": 1,
      "x": 55,
      "y": 30,
      "width": 55,
      "height": 8,
      "fontSize": 10
    },
    {
```

```

"name": "deliveryDate",
"type": "TEXT",
"value": "{{deliveryDate}}",
"page": 1,
"x": 145,
"y": 30,
"width": 47,
"height": 8,
"fontSize": 10
},
{
"name": "customerName",
"type": "TEXT",
"value": "{{customer.name}}",
"page": 1,
"x": 45,
"y": 50,
"width": 147,
"height": 8,
"fontSize": 10
},
{
"name": "customerAddress",
"type": "TEXT",
"value": "{{customer.address}}",
"page": 1,
"x": 45,
"y": 63,
"width": 147,
"height": 8,
"fontSize": 9
},
{
"name": "workorderNumber",
"type": "TEXT",
"value": "{{workorderNumber}}",
"page": 1,
"x": 55,
"y": 87,
"width": 55,
"height": 8,
"fontSize": 10
},
{
"name": "partNumber",
"type": "TEXT",
"value": "{{partNumber}}",
"page": 1,
"x": 55,
"y": 101,
"width": 55,
"height": 8,
"fontSize": 10
},
{
"name": "quantity",
"type": "TEXT",
"value": "{{quantity}}",
"page": 1,
"x": 145,
"y": 101,
"width": 47,
"height": 8,
"fontSize": 10,
"align": "RIGHT"
},
{
"name": "serialNumber",
"type": "TEXT",
"value": "{{serialNumber}}",
"page": 1,
"x": 55,
"y": 115,
"width": 70,
"height": 8,
"fontSize": 10
},
{
"name": "serialBarcode",
"type": "BARCODE",
"barcodeType": "CODE_128",
"value": "{{serialNumber}}",
"page": 1,
"x": 18,
"y": 135,
"width": 100,
"height": 25
},
{
"name": "serialQr",
"type": "QR_CODE",
"value": "{{serialNumber}}",
"page": 1,
"x": 145,
"y": 125,

```

```

"width": 35,
"height": 35
},
{
"name": "note",
"type": "TEXT",
"value": "{{note}}",
"page": 1,
"x": 18,
"y": 205,
"width": 174,
"height": 18,
"fontSize": 9,
"multiline": true
}
]
}

```

Lieferschein

PDF_TEMPLATE Demo - Koordinatenfelder

Lieferschein-Nr.	{{deliveryNumber}}	Datum	{{deliveryDate}}
Kunde	{{customer.name}}		
Adresse	{{customer.address}}		
Arbeitsauftrag	{{workorderNumber}}		
Artikelnummer	{{partNumber}}	Menge	{{quantity}}
Seriennummer	CODE_128 {{serialNumber}}		QR {{serialNumber}}
Bemerkung			
{{note}}			

Diese PDF-Datei ist die feste Hintergrundvorlage. Werte, Barcodes und QR-Codes werden per fields.json gestempelt.

PDF-Grundvorlage ohne Daten.

Lieferschein

Gerendertes Beispiel PDF_TEMPLATE

Lieferschein-Nr.	LS-2026-0001	Datum	09.06.2026
------------------	--------------	-------	------------

Kunde	Muster GmbH
-------	-------------

Adresse	Musterstraße 1, 12345 Musterstadt
---------	-----------------------------------

Arbeitsauftrag	WO-100200
----------------	-----------

Artikelnummer	A123456	Menge	120
---------------	---------	-------	-----

Seriennummer	SNR000001
--------------	-----------



Bemerkung

Lieferung frei Rampe. Bitte Ware bei Annahme prüfen.

Gerendertes Beispiel: Werte, Barcode und QR-Code sind aus sample-data.json erzeugt.

PDF_TEMPLATE gerendert mit JSON-Daten, Barcode und QR-Code.

5.3 DOCX-Vorlage

DOCX-Vorlagen werden in Word oder LibreOffice gestaltet. Normale Platzhalter können im Fließtext stehen. Barcode-/QR-Code-Platzhalter sollten jeweils allein in einem Absatz stehen, damit sie beim Rendern durch Bilder ersetzt werden können.

```
{
  "name": "lieferschein_docx",
  "documentName": "Lieferschein DOCX Vorlage",
  "templateType": "DOCX",
  "contentType": "application/vnd.openxmlformats-officedocument.wordprocessingml.document",
  "sourceFileName": "lieferschein_docx_template.docx",
  "documentFile": "document.docx"
}

{{deliveryNumber}}
{{deliveryDate}}
{{customer.name}}
{{partNumber}}
{{quantity}}
{{serialNumber}}
{{barcode:CODE_128:serialNumber:360x90}}
{{qr:serialNumber}}
```

Lieferschein

DOCX gerendertes Beispiel

Lieferschein-Nr.	LS-2026-0001
Datum	09.06.2026
Kunde	Muster GmbH
Adresse	Musterstraße 1, 12345 Musterstadt
Arbeitsauftrag	WO-100200
Artikelnummer	A123456
Menge	120
Seriennummer	SNR0000001



Bemerkung

Lieferung frei Rampe. Bitte Ware bei Annahme prüfen.

Positionen

Pos.	Artikel	Bezeichnung	Menge
1	A123456	Lieferschein Demo-Artikel	120

Printservice DOCX Demo - LS-2026-0001

Gerendertes DOCX-Beispiel als PDF-Vorschau.

5.4 ZPL/EPL-Vorlage für Etikettendruck

ZPL/EPL ist der schnellste Weg für industrielle Etikettendrucker. Der Service ersetzt die Platzhalter im Text und sendet das Ergebnis per RAW_TCP oder Spooler an den Drucker.

```
^XA
^CF0,35
^F040,40^FDSerial: {{serialNumber}}^FS
^F040,90^FDPart: {{partNumber}}^FS
^F040,145^BCN,80,Y,N,N^FD{{serialNumber}}^FS
^F040,260^BQN,2,6^FDLA,{{serialNumber}}^FS
^XZ
```

6. Authentifizierung und Benutzer/Token

Authentifizierung ist optional. Wenn sie deaktiviert ist, kann jeder Client, der die REST-API erreicht, drucken. Sobald sie aktiviert wird, sendet der Client ein Token über den Header X-Printservice-Token oder über Authorization: Bearer

Scope	Berechtigung
PRINT	Drucken, eigene Queue sehen, eigene Jobs reprinten.
PREVIEW	Vorschau/Validierung ausführen.
ADMIN	Drucker, Vorlagen, Auth, Queue-Einstellungen administrieren.

Scope	Berechtigung
ALL	Alle Scopes zusammen.

REST-Beispiel: User und Token anlegen

```
curl -X POST http://localhost:8080/api/auth/users -H "Content-Type: application/json" -d
@src/main/resources/examples/rest-create-auth-user-request.json
{
  "userName": "line01",
  "displayName": "Linie 01",
  "tokenName": "line01-print-token",
  "enabled": true,
  "scopes": [
    "PRINT",
    "PREVIEW"
  ]
}
```

REST-Beispiel: Auth einschalten

```
{
  "enabled": true,
  "protectPrint": true,
  "protectRenderedDownload": true,
  "protectPreview": false,
  "protectAdmin": true,
  "headerName": "X-Printservice-Token"
}
```

7. Drucker und Aliasverwaltung

WINDOWS_SPOOLER

WINDOWS_SPOOLER nutzt Java Print Service und damit die vom Betriebssystem sichtbaren Drucker. Geeignet für PDF, HTML->PDF, PDF_TEMPLATE und Office-/A4-Drucker.

```
{
  "alias": "OFFICE_A4",
  "type": "WINDOWS_SPOOLER",
  "systemName": "HP_LaserJet_A4",
  "description": "A4 Drucker ueber Windows Spooler"
}
```

RAW_TCP

RAW_TCP sendet Druckdaten direkt an Host/IP und Port, typischerweise Port 9100. Geeignet für ZPL, EPL, ESC/POS, PCL und reine Textprotokolle. Diese Variante umgeht Windows-Spooler-Probleme im Dienstbetrieb.

```
{
  "alias": "LABEL_LINE_01",
  "type": "RAW_TCP",
  "host": "192.168.10.50",
  "port": 9100,
  "language": "ZPL",
  "description": "Zebra Etikettendrucker direkt per TCP"
}
```

8. Drucken, Vorschau und Render-Modi

Render-Modus	Eingabe	Ausgabe/Druckpfad	Typischer Einsatz
AUTO	Vorlage entscheidet	Automatisch	Standardaufrufe
RAW	Text/ZPL/EPL	Direkt an RAW_TCP/Spooler	Etiketten, Maschinen
PDF	HTML	HTML -> XHTML -> PDF -> Druck	Dynamische Dokumente/Tabellen
PDF_TEMPLATE	PDF + fields.json	PDF stempeln -> Druck	Feste A4-Formulare

Render-Modus	Eingabe	Ausgabe/Druckpfad	Typischer Einsatz
DOCX	DOCX	DOCX erzeugen oder DOCX4J_FO -> PDF	Word-Vorlagen

Druckaufruf: einfache PDF-Lieferscheinvorlage

```
{
  "printerAlias": "OFFICE_A4",
  "templateName": "lieferschein_pdf_simple",
  "renderMode": "PDF_TEMPLATE",
  "copies": 1,
  "pdfDpi": 300,
  "includeRenderedBase64": false,
  "data": {
    "deliveryNumber": "LS-2026-0001",
    "deliveryDate": "09.06.2026",
    "customer": {
      "name": "Muster GmbH",
      "address": "Musterstraße 1, 12345 Musterstadt"
    },
    "workorderNumber": "WO-100200",
    "partNumber": "A123456",
    "quantity": "120",
    "serialNumber": "SNR0000001",
    "note": "Lieferung frei Rampe. Bitte Ware bei Annahme prüfen."
  }
}
```

Druckaufruf: HTML-Tabelle

```
{
  "printerAlias": "OFFICE_A4",
  "templateName": "lieferschein_html_table",
  "renderMode": "PDF",
  "copies": 1,
  "pdfDpi": 300,
  "data": {
    "deliveryNumber": "LS-2026-0001",
    "deliveryDate": "08.06.2026",
    "customer": {
      "name": "Muster GmbH"
    },
    "items": [
      {
        "pos": "1",
        "partNumber": "A123456",
        "description": "Gehäuse Oberteil",
        "quantity": "10",
        "serialNumber": "SNR0000001"
      },
      {
        "pos": "2",
        "partNumber": "B987654",
        "description": "Elektronikmodul",
        "quantity": "5",
        "serialNumber": "SNR0000002"
      },
      {
        "pos": "3",
        "partNumber": "C555111",
        "description": "Dichtung",
        "quantity": "20",
        "serialNumber": "SNR0000003"
      }
    ]
  }
}
```

Druckaufruf: DOCX mit DOCX4J_FO

```
{
  "printerAlias": "OFFICE_A4",
  "templateName": "lieferschein_docx",
  "renderMode": "DOCX",
  "docxConverter": "DOCX4J_FO",
  "copies": 1,
  "pdfDpi": 300,
  "includeRenderedBase64": false,
  "data": {
    "deliveryNumber": "LS-2026-0001",
    "deliveryDate": "09.06.2026",
    "customer": {
      "name": "Muster GmbH",
      "address": "Musterstraße 1, 12345 Musterstadt"
    },
    "workorderNumber": "WO-100200",
    "partNumber": "A123456",
    "quantity": "120",

```

```

"serialNumber": "SNR000001",
"note": "Lieferung frei Rampe. Bitte Ware bei Annahme prüfen.",
"description": "Lieferschein Demo-Artikel"
}
}

```

Druckaufruf: RAW_TCP/ZPL

```

{
  "printerAlias": "LABEL_LINE_01",
  "templateName": "serial_label_zpl",
  "renderMode": "RAW",
  "copies": 1,
  "encoding": "UTF-8",
  "data": {
    "serialNumber": "SNR000001",
    "partNumber": "A123456",
    "workorderNumber": "W0-100200"
  }
}

```

9. Druckwarteschlange, Redruck und Datenschutz

Die Queue ist eine eigene persistente Druckhistorie des Printservice und nicht die Windows-Spooler-Warteschlange. Sie kann für Redruck genutzt, gefiltert, bereinigt oder aus Datenschutzgründen deaktiviert werden.

Queue-Einstellungen

```

{
  "enabled": true,
  "storeRequestsForReprint": true,
  "autoCleanupEnabled": true,
  "retentionDays": 30,
  "maxStoredJobs": 5000
}

```

Queue deaktivieren

```

{
  "enabled": false,
  "storeRequestsForReprint": false,
  "autoCleanupEnabled": false,
  "retentionDays": 0,
  "maxStoredJobs": 0
}

```

Redruck

```

curl -X POST http://localhost:8080/api/queue/JOB_ID/reprint -H "Content-Type: application/json" -H "X-Printservice-Token: YOUR_PRINT_TOKEN" -d @src/main/resources/examples/rest-reprint-request.json

```

```

{
  "copies": 1,
  "printerAlias": "LABEL_LINE_01",
  "dataOverride": {
    "reprintReason": "Etikett beschädigt"
  }
}

```

10. Vollständige REST-API-Referenz

Methode	Endpoint	Scope	Beschreibung
GET	/api	frei	Service-Info und Version.
GET	/api/diagnostics/system	frei	System, Java-Version, Konfigurationspfad, Default-Drucker.
GET	/api/printers/diagnostics	frei	Windows-/LocalSystem-/Druckerdiagnose inkl. Empfehlungen.
GET	/api/printers/scan	frei	Spooler/CUPS-Drucker suchen.

Methode	Endpoint	Scope	Beschreibung
GET	/api/printers	frei	Gespeicherte Drucker-Aliase abrufen.
POST	/api/printers	ADMIN	Drucker-Alias anlegen/ändern.
DELETE	/api/printers/{alias}	ADMIN	Drucker-Alias löschen.
GET	/api/templates/storage	frei	Speicherpfade und Ordnerstruktur abrufen.
GET	/api/templates	frei	Vorlagenliste abrufen.
GET	/api/templates/{name}	frei	Vorlage mit Metadaten abrufen.
POST	/api/templates	ADMIN	Vorlage anlegen.
PUT	/api/templates/{name}	ADMIN	Vorlage ändern.
DELETE	/api/templates/{name}	ADMIN	Vorlage löschen.
GET	/api/barcodes/formats	frei	Unterstützte Barcode-Arten abrufen.
GET	/api/auth/status	frei	Auth-Status abrufen.
GET	/api/auth/me	PRINT	Token/User identifizieren.
GET	/api/auth/users	ADMIN	User/Tokenliste abrufen.
POST	/api/auth/users	ADMIN*	User/Token anlegen. *Bei deaktivierter Auth für Initialsetup offen.
DELETE	/api/auth/users/{idOrUserName}	ADMIN	User/Token löschen.
GET	/api/auth/config	ADMIN	Auth-Konfiguration abrufen.
PUT	/api/auth/config	ADMIN	Auth-Konfiguration ändern.
POST	/api/auth/reload	ADMIN	auth.json neu laden.
POST	/api/html/validate	PREVIEW	HTML-Vorlage rendern und XHTML-Normalisierung prüfen.
GET	/api/docx/converters	frei	DOCX-Konverterstatus abrufen.
POST	/api/docx/validate	PREVIEW	DOCX-Konverter mit konkreter Vorlage validieren.
POST	/api/print/preview	PREVIEW	Druckrequest rendern/vorprüfen.
POST	/api/print/preview/html	PREVIEW	Gerendertes HTML für Browser-Vorschau.
POST	/api/print/rendered	PRINT	Gerendertes Dokument herunterladen.
POST	/api/print	PRINT	Druckjob ausführen.
GET	/api/queue	PRINT/ADMIN	Eigene Queue oder mit full=true Admin-Gesamtliste.
GET	/api/queue/{jobId}	PRINT/ADMIN	Einzelnen Queue-Job abrufen.
POST	/api/queue/{jobId}/reprint	PRINT	Job erneut drucken.
GET	/api/queue/settings	ADMIN	Queue-/Datenschutz-Einstellungen abrufen.
PUT	/api/queue/settings	ADMIN	Queue-/Datenschutz-Einstellungen speichern.
DELETE	/api/queue	ADMIN	Queue komplett leeren.
POST	/api/queue/cleanup	ADMIN	Jobs älter als X Tage löschen.

11. cURL-Beispiele

Systemdiagnose

```
curl http://localhost:8080/api/diagnostics/system
```

Drucker scannen

```
curl http://localhost:8080/api/printers/scan
```

Drucker-Alias speichern

```
curl -X POST http://localhost:8080/api/printers -H "Content-Type: application/json" -H  
"X-Printservice-Token: ADMIN_TOKEN" -d  
@src/main/resources/examples/rest-create-raw-tcp-printer-request.json
```

Vorlage speichern

```
curl -X POST http://localhost:8080/api/templates -H "Content-Type: application/json" -H  
"X-Printservice-Token: ADMIN_TOKEN" -d  
@src/main/resources/examples/rest-create-html-table-template-request.json
```

Für PDF- und DOCX-Vorlagen enthält der Create-Template-Request Base64-Inhalt. In der Dokumentation wird der Payload nicht vollständig ausgeschrieben, weil Binärdaten sehr groß sind; die vollständigen Beispiele liegen im Projekt unter `src/main/resources/examples/` und im `config-example/-` Ordner.

Druckjob senden

```
curl -X POST http://localhost:8080/api/print -H "Content-Type: application/json" -H  
"X-Printservice-Token: YOUR_PRINT_TOKEN" -d  
@src/main/resources/examples/rest-print-pdf-simple-lieferschein-request.json
```

Queue filtern

```
# Eigene Jobs  
curl "http://localhost:8080/api/queue?limit=100" -H "X-Printservice-Token:  
YOUR_PRINT_TOKEN"  
  
# Admin: volle Liste, auf User und Vorlage gefiltert  
curl  
"http://localhost:8080/api/queue?full=true&user=line01&templateName=lieferschein_pdf_simple"  
-H "X-Printservice-Token: YOUR_ADMIN_TOKEN"
```

Queue leeren / bereinigen

```
curl -X DELETE http://localhost:8080/api/queue -H "X-Printservice-Token: YOUR_ADMIN_TOKEN"  
  
curl -X POST "http://localhost:8080/api/queue/cleanup?olderThanDays=30" -H "Content-Type:  
application/json" -H "X-Printservice-Token: YOUR_ADMIN_TOKEN" -d "{}"
```

12. Barcode-/QR-Code-Arten und Platzhalter

Barcode- und QR-Code-Erzeugung erfolgt serverseitig. HTML und PDF_TEMPLATE erzeugen Bilder, ZPL/EPL kann native Druckerbefehle nutzen oder Platzhalter ersetzen.

Platzhalter	Bedeutung
{{text:serialNumber}}	Text mit HTML-/XML-sicherem Escaping.
{{serialNumber}}	Einfacher Wert ohne spezielle Formatierung.
{{barcode:CODE_128:serialNumber:360x90}} {	Barcode als Bild, Format CODE_128, Wert aus serialNumber, Größe 360x90.
{{qr:serialNumber}}	QR-Code mit Wert aus serialNumber.
{{data_matrix:serialNumber:120x120}}	DataMatrix-Code mit Größe 120x120.
{{#each:items}}...{/each}}	HTML-Tabellenblock: Wiederholung für jedes Element aus data.items.

Unterstützte Formate können zur Laufzeit über GET `/api/barcodes/formats` abgefragt werden. Typische Formate sind CODE_128, CODE_39, EAN_13, EAN_8, UPC_A, QR_CODE, DATA_MATRIX, PDF_417 und AZTEC.

13. Fehlerbehebung und Produktionshinweise

HTML->PDF XML/SAX-Fehler

openhtmltopdf ist strenger als ein Browser. Ab v2.5 normalisiert der Service HTML zu XHTML, entfernt BOM/ungültige Steuerzeichen und bettet Fragmente in html/body ein. Vorlagen können mit /api/html/validate geprüft werden.

```
{
  "templateName": "lieferschein_html_table",
  "renderMode": "PDF",
  "data": {
    "deliveryNumber": "LS-2026-0001",
    "deliveryDate": "08.06.2026",
    "customer": {
      "name": "Muster GmbH"
    },
    "items": [
      {
        "pos": "1",
        "partNumber": "A123456",
        "description": "Gehäuse Oberteil",
        "quantity": "10",
        "serialNumber": "SNR000001"
      }
    ]
  }
}
```

DOCX4J_FO validieren

DOCX4J_FO ist Java-only und benötigt kein LibreOffice, ist aber bei komplexen Word-Layouts nicht immer 1:1 layouttreu. Jede DOCX-Vorlage sollte vor Produktivdruck validiert werden.

```
{
  "templateName": "lieferschein_docx",
  "renderMode": "DOCX",
  "docxConverter": "DOCX4J_FO",
  "includeRenderedBase64": false,
  "data": {
    "deliveryNumber": "LS-2026-0001",
    "deliveryDate": "09.06.2026",
    "customer": {
      "name": "Muster GmbH",
      "address": "Musterstraße 1, 12345 Musterstadt"
    },
    "workorderNumber": "WO-100200",
    "partNumber": "A123456",
    "quantity": "120",
    "serialNumber": "SNR000001",
    "note": "Lieferung frei Rampe. Bitte Ware bei Annahme prüfen.",
    "description": "Lieferschein Demo-Artikel"
  }
}
```

Windows-Dienst und Drucker

Wenn die Anwendung als Windows-Dienst unter LocalSystem läuft, sind benutzergebundene Drucker oft nicht sichtbar. Empfehlung: serverweit installierte lokale TCP/IP-Drucker für Windows-Spooler oder RAW_TCP für ZPL/EPL-Etikettendrucker verwenden.

Sicherheit und Datenschutz

- Auth für /api/print produktiv aktivieren.
- Für jede Maschine/jeden Client eigenen User und Token verwenden.
- Queue-Retention begrenzen, z.B. 14 oder 30 Tage.
- Bei sensiblen Daten storeRequestsForReprint=false oder Queue deaktivieren.
- Konfigurationsordner sichern, aber auth.json und print-jobs.json mit Berechtigungen schützen.
- Printservice nicht ungeschützt im offenen Netzwerk betreiben.

Anhang: Beispiel-Dateien im Projekt

Beispieldatei	Inhalt/Zweck
src/main/resources/examples/lieferschein_docx_rendered_example.docx	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_docx_rendered_example.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_docx_template.docx	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_html_table.html.tpl	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_html_table_rendered_example.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_pdf_fields.json	REST- oder Template-Beispiel
src/main/resources/examples/lieferschein_pdf_simple_fields.json	REST- oder Template-Beispiel
src/main/resources/examples/lieferschein_pdf_simple_rendered_example.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_pdf_simple_template.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_pdf_table_fields.json	REST- oder Template-Beispiel
src/main/resources/examples/lieferschein_pdf_table_fields_newpage.json	REST- oder Template-Beispiel
src/main/resources/examples/lieferschein_pdf_table_rendered_example.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_pdf_table_template.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/lieferschein_pdf_template.pdf	Vorlagen-/Demo-Datei
src/main/resources/examples/rest-cleanup-queue-older-than-days.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-clear-queue.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-create-admin-user-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-auth-user-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-docx-lieferschein-template-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-html-table-template-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-pdf-simple-lieferschein-template-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-pdf-template-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-pdf-template-table-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-raw-tcp-printer-request.json	Create-/Setup-Request
src/main/resources/examples/rest-create-windows-spooler-printer-request.json	Create-/Setup-Request
src/main/resources/examples/rest-diagnostics-system-response-example.json	REST- oder Template-Beispiel
src/main/resources/examples/rest-disable-queue-for-privacy-request.json	REST- oder Template-Beispiel
src/main/resources/examples/rest-docx-converters.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-get-queue-settings.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-list-queue-filter-user.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-list-queue-full-admin.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-list-queue-own.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-print-docx-lieferschein-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-print-docx4j-fo-request.json	Druckrequest oder cURL-Beispiel

Beispieldatei	Inhalt/Zweck
src/main/resources/examples/rest-print-html-table-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-print-pdf-simple-lieferschein-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-print-pdf-template-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-print-pdf-template-table-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-print-raw-tcp-zpl-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-queue-response-example.json	REST- oder Template-Beispiel
src/main/resources/examples/rest-reprint-request.json	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-reprint.curl.txt	Druckrequest oder cURL-Beispiel
src/main/resources/examples/rest-update-auth-config-enabled-request.json	REST- oder Template-Beispiel
src/main/resources/examples/rest-update-queue-settings-request.json	REST- oder Template-Beispiel
src/main/resources/examples/rest-update-queue-settings.curl.txt	REST- oder Template-Beispiel
src/main/resources/examples/rest-validate-docx-lieferschein-request.json	Validierungsrequest
src/main/resources/examples/rest-validate-docx4j-fo-request.json	Validierungsrequest
src/main/resources/examples/rest-validate-docx4j-fo.curl.txt	Validierungsrequest
src/main/resources/examples/rest-validate-html-pdf-request.json	Validierungsrequest
src/main/resources/examples/serial_label.zpl.tpl	Vorlagen-/Demo-Datei
src/main/resources/examples/serial_label_zpl_barcode_qr.zpl.tpl	Vorlagen-/Demo-Datei
src/main/resources/examples/serienbegleitschein.docx	Vorlagen-/Demo-Datei
src/main/resources/examples/warenanhaenger.html.tpl	Vorlagen-/Demo-Datei
src/main/resources/examples/warenanhaenger_pdf.html.tpl	Vorlagen-/Demo-Datei