

Contents

Document Generation Service - API Contract Specification	3
Executive Summary	4
Service Overview	5
Core Capabilities	5
Service Endpoints Overview	5
API Contract Standards	6
Response Format Standard	6
Status Values	6
Error Format Standard	6
Authentication Contract	7
Authentication Flow	7
Login Request Contract	7
Token Refresh Contract	8
Logout Contract	9
Authentication Status Contract	9
Document Template Management Contract	11
Document Generation Flow	11
Document Template List Contract	11
Document Generation Contract	12
Template Upload Contract	18
Template Details Contract	20
Template Parameters Contract	21
Template Fields Contract	22
ICC Profiles Contract	23
Available Fonts Contract	23
Template Download Contract	24
Template Deletion Contract	25
Template Tags Contract	25
Parameter Types and Validation	29
Supported Parameter Types	29
User Management Contract	30
User List Contract	30
Current User Profile Contract	31
Change User Password Contract	31
Update User Roles Contract	32
Configuration Management Contract	33
Get All Configuration Contract	33
Audit and Monitoring Contract	36
Audit Log Query Contract	36

Audit Statistics Contract	37
User-Specific Audit Logs Contract	38
Security Events Contract	39
Audit System Health Contract	39
Manual Audit Cleanup Contract	39
Audit Dashboard Overview Contract	40
Security Threats Dashboard Contract	40
Compliance Dashboard Contract	41
System Health Metrics Contract	41
File Integrity Management Contract	43
Overview	43
Regenerate Digest Contract	43
Verify Integrity Contract	43
Regenerate All Digests Contract	44
Common Use Cases	45
File Integrity Security	45
Document Post-Processing Contract	47
PDF Post-Processing	47
Async Monitoring API (Optional)	50
Async Metrics Dashboard Contract	50
Async Request List Contract	51
Async Request Details Contract	52
Async Worker Status Contract	53
Async Error Log Contract	54
Async System Health Contract	56
Bulk Async Submission Contract	57
Async Operations API (Optional)	60
Get Async Result Status Contract	60
Download Async Result Contract	61
Acknowledge Async Result Contract	62
Usage Patterns	63
Error Handling Standards	65
Standard Error Codes	65
Validation Error Example	65
Document Generation Error Example	66
Request/Response Headers	67
Required Request Headers	67
Standard Response Headers	67
Integration Guidelines	68
Best Practices	68
Security Considerations	68
Change Management	69
Versioning Strategy	69
Deprecation Policy	69

Document Generation Service - API Contract Specification

Version: 1.15

Document Type: Service Contract Specification

Target Audience: External System Integrators

Classification: Public API Documentation

Last Updated: March 31, 2026

Executive Summary

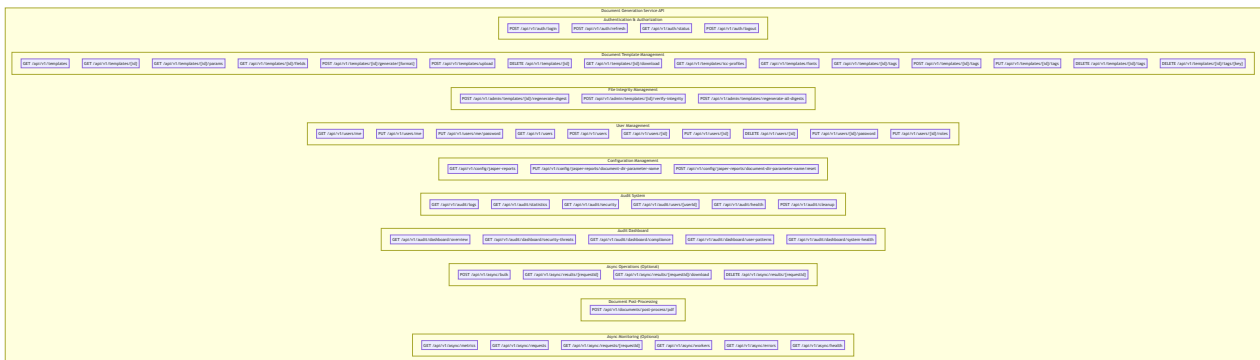
The Document Generation Service provides RESTful API endpoints for automated document generation from structured data using predefined templates. The service supports **dual template engines** — JasperReports (JRXML) and DOCX (Microsoft Word) — with multiple output formats (PDF, XLSX, DOCX, RTF, HTML, TXT) and implements enterprise-grade security, audit trails, and distributed tracing capabilities.

Service Overview

Core Capabilities

- **Template-Based Document Generation:** Transform structured JSON data into formatted documents
- **Dual Template Engines:** JasperReports (JRXML) templates for complex reports with full format support, and DOCX (Microsoft Word) templates for placeholder-based document generation (DOCX, PDF, HTML, and TXT output)
- **Multi-Format Output:** Support for PDF, Microsoft Excel (XLSX), Microsoft Word (DOCX), Rich Text Format (RTF), Plain Text (TXT), and HTML (packaged as ZIP archive)
- **PDF Post-Processing:** Apply encryption, PDF/A conformance, font embedding, image compression, CMYK conversion, transparency flattening, and duplex padding to pre-generated PDF files without re-rendering
- **Automatic Template Type Detection:** The service auto-detects whether an uploaded template is JRXML or DOCX and routes generation to the appropriate engine
- **Template Management:** Upload, validate, and manage document templates
- **Parameter-Driven Generation:** Configurable document parameters with type validation
- **Audit Trail:** Comprehensive logging and tracking of all operations
- **Distributed Tracing:** Request correlation across system boundaries
- **Role-Based Access Control:** Authentication and authorization mechanisms
- **Token Revocation:** Secure logout with refresh token blacklisting and automatic token rotation

Service Endpoints Overview



API Contract Standards

Response Format Standard

All API responses follow a consistent structure to ensure predictable integration:

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "abc123-def456-ghi789",
    "requestId": "req-xyz789",
    "status": "success"
  },
  "data": {
    // Response payload specific to each endpoint
  },
  "errors": [
    // Array of error objects (empty for successful responses)
  ]
}
```

Status Values

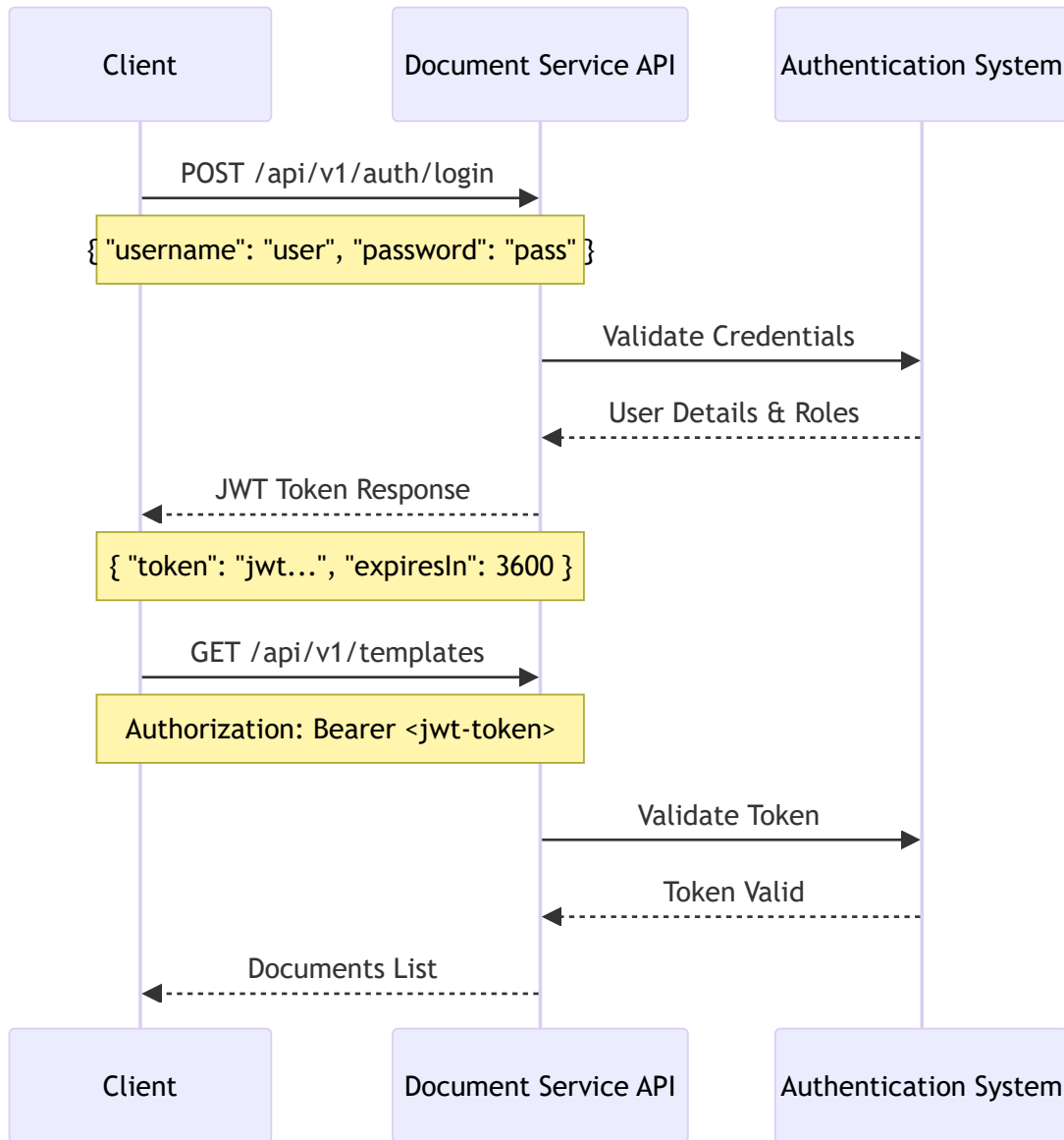
- **success**: Operation completed successfully
- **error**: Operation failed due to client or server error
- **partial**: Operation completed with warnings or partial success

Error Format Standard

```
{
  "meta": {
    "timestamp": "2025-10-10T12:31:00Z",
    "correlationId": "error-456-def789",
    "requestId": "req-err123",
    "status": "error"
  },
  "data": null,
  "errors": [
    {
      "code": "VALIDATION_ERROR",
      "message": "Field validation failed",
      "field": "documentId",
      "details": "Document ID must be a valid UUID"
    }
  ]
}
```

Authentication Contract

Authentication Flow



Login Request Contract

Endpoint: POST /api/v1/auth/login

Request Body:

```

{
  "username": "string",
  "password": "string"
}
    
```

Successful Response:

```

{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "auth-123-abc",
    "requestId": "req-login-001",
    "status": "success"
  },
  "data": {
    "accessToken": "eyJhbGciOiJIUzUxMiJ9...",
    "refreshToken": "eyJhbGciOiJIUzUxMiJ9...",
    "tokenType": "Bearer",
    "expiresIn": 3600,
    "userId": "550e8400-e29b-41d4-a716-446655440000"
  },
  "errors": []
}

```

Response Fields:

- accessToken: JWT access token for API authentication (1 hour validity by default)
- refreshToken: JWT refresh token for obtaining new access tokens (24 hours validity by default)
- tokenType: Always "Bearer"
- expiresIn: Access token validity in seconds (per OAuth 2.0 RFC 6749)
- userId: UUID of the authenticated user

Token Refresh Contract

Endpoint: POST /api/v1/auth/refresh

Description: Obtain a new access token and refresh token using a valid refresh token. The old refresh token is automatically revoked (token rotation) to prevent replay attacks.

Request Headers:

- Authorization: Bearer <refresh_token> (required)

Successful Response:

```

{
  "meta": {
    "timestamp": "2026-01-16T12:30:00Z",
    "correlationId": "refresh-789-xyz",
    "requestId": "req-refresh-001",
    "status": "success"
  },
  "data": {
    "accessToken": "eyJhbGciOiJIUzUxMiJ9...(new)",
    "refreshToken": "eyJhbGciOiJIUzUxMiJ9...(new)",
    "tokenType": "Bearer",
    "expiresIn": 3600,
    "userId": "550e8400-e29b-41d4-a716-446655440000"
  },
  "errors": []
}

```

Error Responses:

- 401 TOKEN_REVOKED: The refresh token has been revoked (logged out or rotated)
- 401 INVALID_TOKEN_TYPE: An access token was provided instead of a refresh token
- 401 INVALID_TOKEN: The token is malformed or expired

Logout Contract

Endpoint: POST /api/v1/auth/logout

Description: Logout the user by revoking the provided refresh token. After logout, the refresh token cannot be used to obtain new access tokens. Access tokens remain valid until they expire (stateless by design).

Request Headers:

- Authorization: Bearer <refresh_token> (required)

Successful Response:

```
{
  "meta": {
    "timestamp": "2026-01-16T12:35:00Z",
    "correlationId": "logout-456-abc",
    "requestId": "req-logout-001",
    "status": "success"
  },
  "data": {
    "tokenRevoked": true,
    "message": "Logout successful",
    "username": "integrator_user"
  },
  "errors": []
}
```

Error Responses:

- 400 INVALID_TOKEN_TYPE: An access token was provided instead of a refresh token
- 401 INVALID_TOKEN: The token is malformed or expired

Security Notes:

- Refresh tokens are blacklisted in the database and cannot be reused
- Token rotation ensures that each refresh token can only be used once
- Blacklist entries are automatically cleaned up after token expiration
- All logout events are recorded in the audit log

Authentication Status Contract

Endpoint: GET /api/v1/auth/status

Description: Returns authentication configuration status including all enabled authentication methods. Includes both legacy flags (for backward compatibility) and current configuration flags.

Response:

```
{
  "meta": {
    "timestamp": "2025-11-21T08:14:26Z",
    "correlationId": "status-456-def",
    "requestId": "req-status-001",
  }
}
```

```
  "status": "success"
},
"data": {
  "oauth2Enabled": true,
  "jwtEnabled": true,
  "adfsEnabled": true,
  "internalEnabled": true,
  "oauth2ClientEnabled": false,
  "oauth2WebEnabled": false
},
"errors": []
}
```

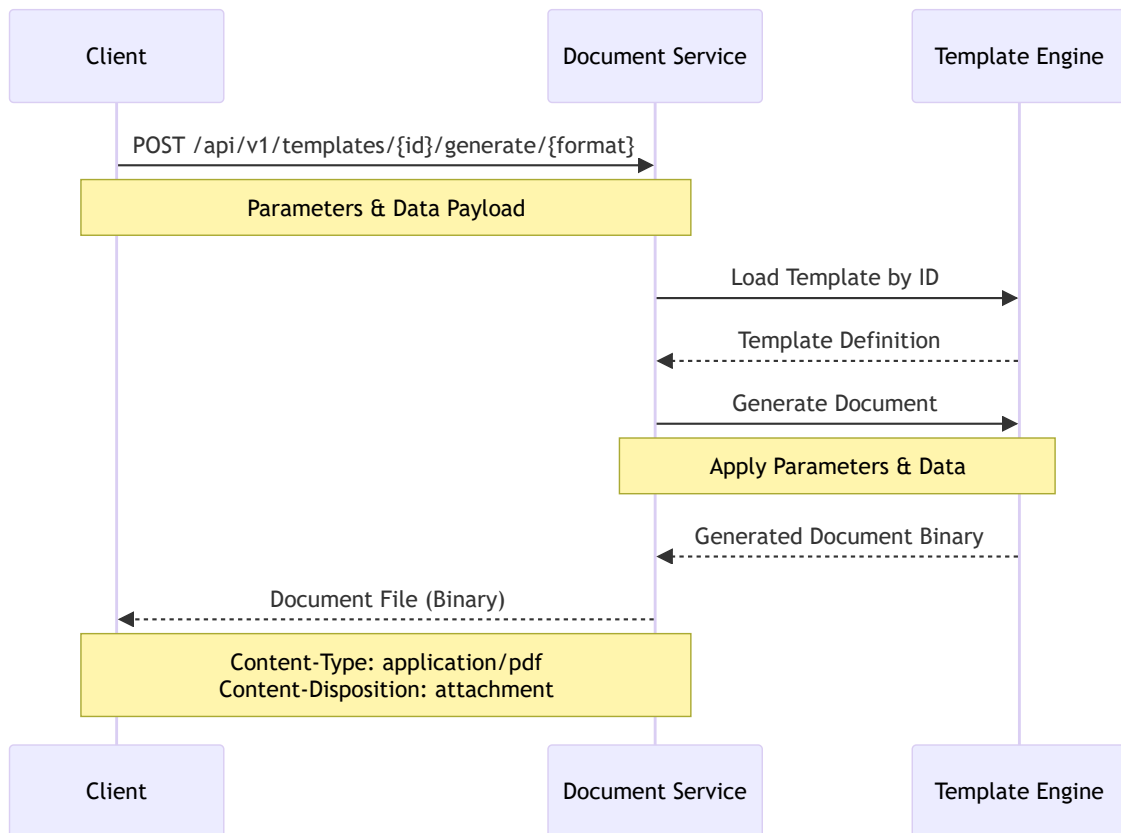
Response Fields:

- `oauth2Enabled`, `jwtEnabled`, `adfsEnabled`: Legacy flags for backward compatibility
- `internalEnabled`: Internal JWT authentication status (`auth.internal.enabled`)
- `oauth2ClientEnabled`: OAuth2 client mode status (`auth.oauth2.client.enabled`)
- `oauth2WebEnabled`: OAuth2 web authentication status (`auth.oauth2.web.enabled`)

Note: Multiple flags can be true simultaneously — all three authentication methods are additive and can coexist in the same deployment.

Document Template Management Contract

Document Generation Flow



Document Template List Contract

Endpoint: GET `/api/v1/templates`

Query Parameters:

- `page` (optional): Page number for pagination (default: 1, 1-based)
- `size` (optional): Page size (default: 20, max: 100)
- `search` (optional): Broad search term across name, description, and metadata (OR logic)
- `name` (optional): Filter by template name specifically (AND logic with other filters)
- `description` (optional): Filter by template description specifically (AND logic with other filters)
- `metadata` (optional): Filter by template metadata specifically (AND logic with other filters)
- `author` (optional): Filter by author name (AND logic with other filters)
- `createdAfter` (optional): Filter templates created after this date (ISO 8601 format)
- `createdBefore` (optional): Filter templates created before this date (ISO 8601 format)
- `minFileSize` (optional): Filter templates with file size \geq this value (in bytes)
- `maxFileSize` (optional): Filter templates with file size \leq this value (in bytes)
- `sortBy` (optional): Field to sort by (name, author, created, fileSize). Default: created
- `sortDir` (optional): Sort direction (asc, desc). Default: desc
- `exactMatch` (optional): If true, all text filters (search, name, description, metadata, author) use exact matching instead of partial LIKE matching. Default: false

- `templateType` (optional): Filter by template type — JASPER or DOCX (case-insensitive). Omit to return all types.
- `tags` (optional): Filter by template tags in `key:value` format. Multiple tags are combined with AND logic. Use `key:` (without value) to match any value for a given key. Example: `tags=phase:prod&tags=department:finance`

Filter Behavior:

- `search`: Matches templates where name, description, OR metadata contains the term
- `name, description, metadata, author`: These are specific field filters that combine with AND logic
- `tags`: Each tag requires a matching entry in the template's tag set; multiple tags use AND logic
- All text filters are case-insensitive

Response: ApiResponseDto<PaginatedDocumentsDto>

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "list-789-ghi",
    "requestId": "req-list-001",
    "status": "success"
  },
  "data": {
    "totalItems": 45,
    "totalPages": 3,
    "currentPage": 1,
    "itemsPerPage": 20,
    "items": [
      {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "name": "Invoice Template",
        "description": "Standard invoice template for billing",
        "metadata":
          ↵ [{"tags":["finance","billing"],"version":"1.0","department":"Accounting"}],
        "author": "System Administrator",
        "created": "2025-10-01T10:00:00Z",
        "templateType": "JASPER",
        "tags": [
          { "key": "department", "value": "accounting" },
          { "key": "phase", "value": "prod" }
        ]
      }
    ]
  },
  "errors": []
}
```

Document Generation Contract

Endpoint: POST `/api/v1/templates/{id}/generate/{format}`

Path Parameters:

- `id`: UUID of the document template
- `format`: Output format (PDF, XLSX, DOCX, RTF, HTML, TXT)
 - HTML: Returns a ZIP archive containing `index.html` and associated assets

- **DOCX templates** support DOCX, PDF, HTML, and TXT output formats. Requesting XLSX or RTF from a DOCX template returns 400 UNSUPPORTED_FORMAT

Request Body:

```
{
  "parameters": [
    {
      "name": "customerName",
      "value": "Acme Corporation Ltd."
    },
    {
      "name": "invoiceDate",
      "value": "2025-10-10"
    },
    {
      "name": "totalAmount",
      "value": 6000.00
    },
    {
      "name": "isVipCustomer",
      "value": true
    },
    {
      "name": "invoiceNumber",
      "value": 12345
    }
  ],
  "data": {
    "items": [
      {
        "description": "Professional Services",
        "quantity": 40,
        "unitPrice": 150.00,
        "total": 6000.00
      },
      {
        "description": "Consulting Services",
        "quantity": 20,
        "unitPrice": 200.00,
        "total": 4000.00
      }
    ],
    "companyLogo": "base64-encoded-image-data",
    "companyAddress": {
      "street": "123 Business Ave",
      "city": "Business City",
      "postalCode": "12345",
      "country": "Business Country"
    }
  },
  "filename": "invoice-october-2025",
  "documentLocale": "en_US",
  "ignorePagination": false,
  "pdfExportOptions": {
    "iccProfile": "pso-coated-v3",
    "pdfaConformance": "PDF/A-1b"
  }
}
```

}

Request Fields:

- **parameters** (required for JRXML templates / optional for DOCX templates): Array of parameter name-value pairs. For JRXML templates, these are Jaspersoft report parameters passed as `$P{parameterName}`. For DOCX templates, these values replace `${placeholder}` markers in the document. Values can be provided in native JSON types (numbers, booleans) or as strings for backward compatibility
- **data** (optional): Datasource objects and complex data structures for report processing (collections, tables, images). For DOCX templates, nested data is accessed via dot-notation using SpEL navigation (e.g., `{"address": {"city": "Warsaw"}}` resolves `${address.city}`)
- **filename** (optional): Custom filename for the generated document (without extension)
- **documentLocale** (optional): Locale for internationalization and formatting (e.g., "pl_PL", "en_US", "de_DE")
- **ignorePagination** (optional): Controls pagination behavior in the generated document. When true, ignores pagination and renders as continuous content. When false (default), uses normal pagination with page breaks. Useful for web display or continuous content flow
- **pdfExportOptions** (optional): PDF-specific export options. Only applies when format is PDF, ignored for XLSX, DOCX, RTF, HTML and TXT
 - **iccProfile** (optional): ICC color profile name for CMYK color management (e.g., "pso-coated-v3", "coated-fogra39"). Profile files must be available in `src/main/resources/icc/` directory. Use with `pdfaConformance` to embed profile in `OutputIntent`
 - **pdfaConformance** (optional): PDF/A conformance level for archival purposes. Currently supports "PDF/A-1b" only. Adds XMP metadata and `OutputIntent` with ICC profile using Apache PDFBox post-processing. Note: JasperReports 7.x with OpenPDF does NOT natively support PDF/A - this feature uses post-processing to add compliance. **Important:** PDF/A and encryption are mutually exclusive (ISO 19005)
 - **userPassword** (optional): Password required to open and view the PDF document. When set, the document will be encrypted with AES-128 (default) or AES-256
 - **ownerPassword** (optional): Password required to change security settings or permissions. If only `ownerPassword` is set, document opens without password but has restrictions
 - **canPrint** (optional, default: true): Allow printing the document
 - **canPrintHighQuality** (optional, default: true): Allow high-resolution printing
 - **canModify** (optional, default: false): Allow modifying document content
 - **canCopy** (optional, default: true): Allow copying text and images
 - **canFillForms** (optional, default: true): Allow filling form fields
 - **canAnnotate** (optional, default: true): Allow adding annotations
 - **canAssemble** (optional, default: false): Allow inserting, rotating, or deleting pages
 - **encryptionKeyLength** (optional, default: 128): Encryption key length (128 or 256 bits)
 - **duplexPadding** (optional, default: false): Pad document to an even number of pages for duplex (double-sided) printing. When true, appends a blank page if the page count is odd. The blank page matches the size of the last page. Compatible with PDF/A conformance and security options

- `fontEmbeddingSubstitute` (optional): Font family name to substitute for non-embedded base-14 fonts (Helvetica, Courier, Times). When set, all non-embedded Type 1 fonts in the generated PDF are replaced with the specified embedded TrueType font. The font must be available — either bundled in the template package (via `fonts.xml`) or installed on the server. Leave empty or omit to skip font embedding post-processing. **Examples:** “Arial”, “DejaVu Sans”
- `imageCompressionQuality` (optional): JPEG quality (0.0–1.0) for post-processing image compression. When set, large lossless-encoded images (FlateDecode / raw) in the PDF are re-compressed as JPEG with the specified quality level. Only colour RGB images above a minimum pixel threshold are converted; small icons, grayscale masks, and already-JPEG images are left untouched. **0.85** — good quality, significant size reduction (recommended for print). **0.75** — acceptable for mass printing, aggressive compression. **0.95** — near-lossless, moderate reduction. Leave empty or omit to skip image optimisation
- `cmkConversionProfile` (optional): ICC profile name for RGB→CMYK conversion. When set, RGB raster images are converted to DeviceCMYK colour space using the specified ICC profile, and RGB colour operators in page/form content streams are rewritten to CMYK operators so text and vector elements also use CMYK output. Advanced colour spaces (e.g., Pattern, Separation, DeviceN) remain unchanged. Available profiles: **coated-fogra39** (ISO Coated v2, general commercial print), **ps0-coated-v3** (PSO Coated v3, modern offset), **ps0-uncoated-v3** (PSO Uncoated v3, uncoated paper). Leave empty or omit to skip CMYK conversion
- `cmkPreserveBlack` (optional, default: true): Use *Preserve Black* during RGB→CMYK conversion. When true, achromatic RGB sources (R=G=B in vector content; matching pixels in raster images) are mapped to **K-only CMYK** (C=0, M=0, Y=0, K=1-L) instead of the 4-colour composite black an ICC transform would produce. Matches the behaviour of professional tools (Acrobat *Preserve Black*, Ghostscript `-dPreserveBlack=true`, RIP “Black Text in K”) — text is sharper on press, ink consumption for black areas drops to ~25 %, no misregistration on small glyphs. Strict R=G=B equality is required, so photographic content (which always has chromatic noise) still goes through the full ICC transform; only synthetic content (text, logos, frames, screenshots) is affected. Set to false for proof simulation or when the receiving workflow expects ICC-faithful composite black. Has no effect when `cmkConversionProfile` is empty
- `flattenTransparency` (optional, default: false): Remove transparency from PDF for print-ready output. When true, a two-step process is applied: **(1) Soft-mask flattening** — images with an `/SMask` (alpha channel from PNG sources) are composited onto a white background and the soft mask is removed, eliminating per-image transparency; **(2) Page group removal** — each page is inspected for remaining transparency usage (`SMask`, `BlendMode`, `CA/ca`) and pages that declare a `/Group` `/Transparency` entry but use no real transparency have the entry removed. This produces fully opaque PDFs that print-shop RIPs (Raster Image Processors) can process without activating the transparency blending pipeline. Has no visible effect on the document appearance for documents designed with white/light backgrounds
- `txtExportOptions` (optional): TXT-specific export options. Only applies when format is TXT, ignored for other formats
 - `characterWidth` (optional, default: 8.0): Character width in pixels used for text layout calculations. Controls horizontal character spacing in output. **JasperReports only.**
 - `characterHeight` (optional, default: 13.948): Character height in pixels used for text layout calculations. Controls vertical line spacing in output. **JasperReports only.**

- `pageWidthInChars` (optional): Page width measured in characters. For JasperReports: overrides `characterWidth` for horizontal layout. For DOCX: soft word-wrap width — lines are broken at spaces so no line exceeds this limit
- `pageHeightInChars` (optional): Page height measured in characters. When set, overrides `characterHeight` for vertical layout. **JasperReports only.**
- `lineSeparator` (optional): Custom line separator string (e.g., `“\n”`, `“\r\n”`). Applies to both JasperReports and DOCX templates
- `pageSeparator` (optional): Custom page separator string inserted between pages. **JasperReports only.**
- `trimLineRight` (optional, default: `false`): Whether to trim trailing whitespace from each line. Applies to both JasperReports and DOCX templates

Parameter Value Formats:

Parameters support both native JSON types and string representations:

```
// Native JSON types (recommended)
{
  "name": "totalAmount",
  "value": 1250.75
}

// String representation (backward compatible)
{
  "name": "totalAmount",
  "value": "1250.75"
}

// Boolean values
{
  "name": "isActive",
  "value": true // or "true" as string
}
```

Optional Type Override:

Parameters support an optional `type` field that gives the API caller explicit control over the Java type used by the template engine. This is useful for S2S integrations that need deterministic type resolution instead of relying on Jackson’s JSON type inference (e.g. JSON 2 may deserialize as `Integer` or `Double` depending on context).

```
// Force BigDecimal for financial precision
{
  "name": "totalAmount",
  "value": 15750.25,
  "type": "BigDecimal"
}

// Force Integer even when JSON value is a decimal
{
  "name": "itemCount",
  "value": 5.0,
  "type": "Integer"
}
```

Supported type names (case-insensitive): String, Integer, Long, Double, Float, Boolean, BigDecimal, Decimal, Date, LocalDate, LocalDateTime, Timestamp. Fully-qualified Java class names are also accepted (e.g. java.math.BigDecimal).

- For **DOCX** templates: the override is applied before SpEL evaluation
- For **Jasper** templates: the override takes precedence over the template-declared type
- When **omitted**: current behavior is preserved — fully backward compatible

Optional Format Pattern (DOCX only):

Parameters support an optional format field that controls the output shape of numeric and date values in DOCX templates. This is applied after SpEL evaluation, giving the caller control over number formatting (decimal places, grouping) and date formatting independently of the template.

```
// Format number with grouping and 2 decimal places
{
  "name": "totalAmount",
  "value": 15750.25,
  "type": "BigDecimal",
  "format": "#,##0.00"
}

// Format date as dd.MM.yyyy
{
  "name": "issueDate",
  "value": "2025-03-15",
  "type": "LocalDate",
  "format": "dd.MM.yyyy"
}
```

- For numbers: uses DecimalFormat syntax (e.g. #,##0.00, 0.###)
- For dates: uses DateTimeFormatter syntax (e.g. dd.MM.yyyy, yyyy-MM-dd HH:mm)
- The documentLocale controls locale-sensitive symbols (decimal separator, grouping separator, month names); the format pattern controls the shape
- Applied only to **simple key** placeholders (\${amount}), not SpEL expressions
- **Ignored** for Jasper templates — use .jxml pattern attributes instead
- When **omitted**: the evaluator's default formatting is used

Important Note on Parameters vs Data:

- **Parameters section**: Contains report parameters - simple scalar values (STRING, INTEGER, DECIMAL, BOOLEAN, DATE, DATETIME). For JRXML templates, these are passed directly to the report engine as \$P{parameterName}. For DOCX templates, they replace \${parameterName} placeholders
- **Data section**: Contains datasource collections and complex objects that populate report tables, subreports, and other data-driven components. Arrays of objects should always be placed here, not in parameters. For DOCX templates, nested data is flattened to dot-notation keys
- **Example**: Customer name goes in parameters, but invoice line items array goes in data

Successful Response:

HTTP Status: 200 OK

Response Headers:

- Content-Type: Format-specific MIME type

- PDF: application/pdf
- XLSX: application/vnd.openxmlformats-officedocument.spreadsheetml.sheet
- DOCX: application/vnd.openxmlformats-officedocument.wordprocessingml.document
- RTF: application/rtf
- Content-Disposition: attachment; filename="<document-name>.<extension>"
- X-Processing-Time-Ms: Server-side processing time in milliseconds (excludes download time)

Response Body: Binary document content (actual generated file)

Example Response Headers:

```
HTTP/1.1 200 OK
Content-Type: application/pdf
Content-Disposition: attachment; filename="invoice-20251010-123500.pdf"
Content-Length: 245760
X-Correlation-ID: gen-abc-123
X-Request-ID: req-generate-001
X-Processing-Time-Ms: 1523
```

[Binary PDF content...]

Template Upload Contract

Endpoint: POST /api/v1/templates/upload

Content-Type: multipart/form-data

Authorization: Requires ROLE_MANAGER authority

Form Fields:

Field	Required	Max Length	Description
name	Yes	255 chars	Template name
description	No	1000 chars	Human-readable template description. If omitted, the service auto-extracts it from the template file (JRXML report property <code>com.jaspersoft.studio.report.description</code> or DOCX Dublin Core <code>dc:description</code>)
metadata	No	TEXT	Template metadata. Can be used as a flexible metadata field for storing structured information such as JSON or XML strings for versioning, configuration flags, business rules, or integration context
author	Yes	255 chars	Template author name
file	Yes	N/A	Template package: either a ZIP file containing JRXML and resources, or a standalone .docx Word template with <code>placeholder</code> syntax

Response: ApiResponseDto<DocumentDto>

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "upload-123-abc",
    "requestId": "req-upload-001",
    "status": "success"
  },
  "data": {
```

```

    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Monthly Report Template",
    "description": "Template for generating monthly business reports",
    "metadata": "{ \"tags\": [\"finance\", \"monthly\", \"reports\"], \"version\": \"2.1\", \"department\": \"Accounting\", \"schedule\": \"monthly\" }",
    "author": "Integration Team",
    "created": "2025-10-10T14:30:00Z",
    "templateType": "JASPER"
  },
  "errors": []
}

```

Description Auto-Extraction:

If the description field is not provided (or is blank) during upload, the service attempts to auto-extract it from the template file:

- **JRXML templates:** Reads the `com.jaspersoft.studio.report.description` report property from the compiled JasperDesign
- **DOCX templates:** Reads the Dublin Core `dc:description` from the OPC core properties (File → Properties → Description in Microsoft Word)

If auto-extraction finds a value, it is saved as the template description. A user-provided description always takes precedence.

Metadata Usage Example:

The metadata field is a free-form TEXT field. Structured JSON is recommended for advanced use cases such as tagging, versioning, and configuration:

```

{
  "metadata": "{ \"tags\": [\"finance\", \"quarterly\", \"compliance\"], \"version\": \"2.1.0\", \"department\": \"Accounting\", \"schedule\": \"quarterly\", \"outputFormats\": [\"PDF\", \"XLSX\"], \"owner\": \"finance-team@example.com\" }"
}

```

Parsed, this represents:

```

{
  "tags": ["finance", "quarterly", "compliance"],
  "version": "2.1.0",
  "department": "Accounting",
  "schedule": "quarterly",
  "outputFormats": ["PDF", "XLSX"],
  "owner": "finance-team@example.com"
}

```

The service treats metadata as an opaque string — the JSON structure is a convention for consuming applications to parse and use for filtering, categorization, or integration logic.

Error Scenarios:

The endpoint returns HTTP 422 (Unprocessable Entity) with `INVALID_TEMPLATE_PACKAGE` error code for:

- **Invalid ZIP format:** File is not a valid ZIP archive and is not a valid `.docx` file

```
{
  "errors": [{
    "code": "INVALID_TEMPLATE_PACKAGE",
    "message": "The uploaded file is not a valid ZIP archive. Please ensure you're uploading a
    ↪ properly formatted ZIP file."
  }]
}
```

- **Empty or corrupted ZIP:** ZIP file contains no entries or is corrupted

```
{
  "errors": [{
    "code": "INVALID_TEMPLATE_PACKAGE",
    "message": "The uploaded ZIP file is empty or corrupted. Please upload a valid ZIP file
    ↪ containing a JRXML template."
  }]
}
```

- **Missing JRXML file:** ZIP file doesn't contain any .jrxml files (and is not a DOCX template)

```
{
  "errors": [{
    "code": "INVALID_TEMPLATE_PACKAGE",
    "message": "No JRXML file found in template package. The ZIP file must contain at least one
    ↪ .jrxml file."
  }]
}
```

Notes:

- The service **automatically detects the template type** based on the uploaded file:
 - **ZIP files containing .jrxml:** Classified as JASPER templates. Parameters are extracted from the JRXML file. All output formats (PDF, XLSX, DOCX, RTF, HTML, TXT) are supported
 - **Standalone .docx files:** Classified as DOCX templates. Placeholders (`${name}`) are extracted from document body, headers, and footers. Only PDF and DOCX output formats are supported
- The `templateType` field in responses indicates which template engine will be used: JASPER or DOCX
- The service processes the uploaded file to extract parameters, no manual parameter definition needed
- The service **auto-extracts the template description** if not provided by the user (from JRXML report properties or DOCX core properties)
- File size limit: 50MB
- ZIP file validation includes format checking, corruption detection, and JRXML file presence verification

Template Details Contract

Endpoint: GET `/api/v1/templates/{id}`

Path Parameters:

- `id`: UUID of the template

Response: `ApiResponseDto<DocumentDto>`

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
```

```

    "correlationId": "doc-456-def",
    "requestId": "req-doc-001",
    "status": "success"
  },
  "data": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "name": "Invoice Template",
    "description": "Standard invoice template for billing",
    "metadata":
      ↪ [{"tags":["finance","billing"],"version":"1.0","department":"Accounting"}],
    "author": "System Administrator",
    "created": "2025-10-01T10:00:00Z",
    "templateType": "JASPER",
    "tags": [
      { "key": "department", "value": "accounting" },
      { "key": "phase", "value": "prod" }
    ]
  },
  "errors": []
}

```

Template Parameters Contract

Endpoint: GET /api/v1/templates/{id}/params

Path Parameters:

- id: UUID of the template

Response: ApiResponseDto<List<DocumentParameterDto>>

```

{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "params-789-abc",
    "requestId": "req-params-001",
    "status": "success"
  },
  "data": [
    {
      "name": "customerName",
      "type": "String",
      "defaultValue": null,
      "description": "Customer full name"
    },
    {
      "name": "invoiceDate",
      "type": "Date",
      "defaultValue": null,
      "description": "Invoice generation date"
    }
  ],
  "errors": []
}

```

Template Fields Contract

Endpoint: GET /api/v1/templates/{id}/fields

Path Parameters:

- id: UUID of the template

Description: Retrieves field definitions from JasperReports template. Fields define the data structure expected by the report and are accessed using `#{fieldName}` expressions in the template. **Note:** This endpoint is only applicable to JRXML (JASPER) templates — DOCX templates do not have field definitions.

Response: ApiResponseDto<List<TemplateFieldDto>>

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "fields-456-def",
    "requestId": "req-fields-001",
    "status": "success"
  },
  "data": [
    {
      "name": "customerName",
      "type": "String",
      "description": "Customer full name"
    },
    {
      "name": "orderDate",
      "type": "Date",
      "description": "Order placement date"
    },
    {
      "name": "totalAmount",
      "type": "Decimal",
      "description": "Total order amount"
    }
  ],
  "errors": []
}
```

Notes:

- Field definitions are automatically extracted from the JRXML template using JasperReports API
- Fields represent the expected data structure for report datasources
- Field types correspond to Java class types: String, Number, Decimal, Boolean, Date, or the original class name
- **Field types are informational only** - the service does not validate the data payload against field definitions
- JasperReports handles type conversion at runtime when filling the report
- This information helps clients understand what data fields should be provided in the data section during document generation

ICC Profiles Contract

Endpoint: GET /api/v1/templates/icc-profiles

Description: Retrieves list of available ICC color profiles installed in the system. These profiles can be used with the `pdfExportOptions.iccProfile` parameter when generating PDF documents to ensure accurate CMYK color reproduction for professional printing.

Response: `ApiResponseDto<List<String>>`

```
{
  "meta": {
    "timestamp": "2025-10-28T12:30:00Z",
    "correlationId": "icc-789-abc",
    "requestId": "req-icc-001",
    "status": "success"
  },
  "data": [
    "coated-fogra39",
    "pso-coated-v3",
    "pso-uncoated-v3"
  ],
  "errors": []
}
```

Notes:

- Profile names are returned without the `.icc` file extension
- Profiles are automatically discovered from the system's ICC profile directory
- The returned profile names can be used directly in the `pdfExportOptions.iccProfile` field during PDF generation
- Common profiles include:
 - `coated-fogra39`: FOGRA39 standard for coated paper (widely used European standard)
 - `pso-coated-v3`: FOGRA51 standard for coated paper (latest European standard)
 - `pso-uncoated-v3`: FOGRA52 standard for uncoated paper

Available Fonts Contract

Endpoint: GET /api/v1/templates/fonts

Description: Retrieves list of all font families available for document generation. This includes both system fonts bundled with the application and dynamically loaded fonts from template packages.

Response: `ApiResponseDto<List<FontFamilyDto>>`

```
{
  "meta": {
    "timestamp": "2025-12-18T12:30:00Z",
    "correlationId": "fonts-abc-123",
    "requestId": "req-fonts-001",
    "status": "success"
  },
  "data": [
    {
      "name": "DejaVu Sans",
      "pdfEncoding": "Identity-H",
    }
  ]
}
```

```

    "pdfEmbedded": true,
    "availableStyles": ["normal", "bold", "italic", "boldItalic"],
    "source": "SERVICE"
  },
  {
    "name": "Arial",
    "pdfEncoding": "Identity-H",
    "pdfEmbedded": true,
    "availableStyles": ["normal", "bold"],
    "source": "SERVICE"
  },
  {
    "name": "Merriweather Sans",
    "pdfEncoding": "Identity-H",
    "pdfEmbedded": true,
    "availableStyles": ["normal", "bold", "italic", "boldItalic"],
    "source": "TEMPLATE"
  }
],
"errors": []
}

```

FontFamilyDto Structure:

Field	Type	Description
name	String	Font family name as used in JRXML templates
pdfEncoding	String	PDF encoding (typically "Identity-H" for Unicode)
pdfEmbedded	Boolean	Whether font is embedded in PDF output
availableStyles	Array	Available font styles: normal, bold, italic, boldItalic
source	Enum	Font origin: SERVICE (built-in/service-configured) or TEMPLATE (loaded from template package) or SYSTEM (OS-installed)

Notes:

- Font names returned can be used directly in JasperReports `fontName` attribute
- All fonts with `pdfEmbedded: true` are suitable for PDF/A compliant documents
- System fonts (bundled) are always available (`source: SERVICE`)
- Template-specific fonts are loaded dynamically from `fonts.xml` in template packages (`source: TEMPLATE`)
- **Security:** Font override protection is template-type-aware. `SERVICE` fonts (built-in) are always protected. For `DOCX` templates, `SYSTEM` fonts (OS-installed) are also protected since the FOP pipeline already has access to them. `JASPER` templates may bundle system fonts because JasperReports cannot access OS fonts directly
- Use this endpoint to verify font availability before document generation

Template Download Contract

Endpoint: GET `/api/v1/templates/{id}/download`

Path Parameters:

- `id`: UUID of the document template

Response: Binary ZIP file containing template and resources

HTTP Status: 200 OK

Response Headers:

- Content-Type: application/zip
- Content-Disposition: attachment; filename="template.zip"

Response Body: Binary ZIP content (actual template file)

Template Deletion Contract

Endpoint: DELETE /api/v1/templates/{id}

Path Parameters:

- id: UUID of the template to delete

Authorization: Requires ROLE_MANAGER authority

Response: ApiResponseDto<Void>

HTTP Status: 204 No Content

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "delete-456-def",
    "requestId": "req-delete-001",
    "status": "success"
  },
  "data": null,
  "errors": []
}
```

Note: Following RESTful best practices, DELETE operations return 204 No Content to indicate successful deletion without response body content.

Template Tags Contract

Template tags are mutable key-value operational attributes (e.g. lifecycle phase, department assignment) that exist independently of the immutable template content. They allow organizing and filtering templates without modifying the template itself.

Get Template Tags

Endpoint: GET /api/v1/templates/{id}/tags

Path Parameters:

- id: UUID of the template

Authorization: Requires authentication (isAuthenticated())

Response: ApiResponseDto<List<Tag>>

```
{
  "meta": {
    "timestamp": "2026-03-28T12:00:00Z",
```

```

    "correlationId": "tags-123-abc",
    "requestId": "req-tags-001",
    "status": "success"
  },
  "data": [
    { "key": "phase", "value": "prod" },
    { "key": "department", "value": "finance" },
    { "key": "campaign", "value": "Q3-2026" }
  ],
  "errors": []
}

```

Error Responses:

HTTP Status	Description
401	Unauthorized — missing or invalid credentials
404	Template not found

Replace Template Tags

Endpoint: PUT /api/v1/templates/{id}/tags

Path Parameters:

- id: UUID of the template

Authorization: Requires ROLE_MANAGER authority

Request Body:

```

{
  "tags": [
    { "key": "phase", "value": "prod" },
    { "key": "department", "value": "finance" }
  ]
}

```

Tag Validation Rules:

Field	Constraint	Pattern
key	Required, max 64 chars	^[a-zA-Z0-9_-]+\$
value	Required, max 255 chars	^[a-zA-Z0-9_.\-]+\$
tags	Required, max 20 items	—

Behavior: This is an idempotent replace operation — the complete desired tag set is sent. Any existing tags not in the request are removed. Use POST /{id}/tags for concurrent-safe additive operations.

Response: ApiResponseDto<List<Tag>> — returns the saved tag set (same format as GET).

Error Responses:

HTTP Status	Description
400	Validation error (blank key, invalid characters, too many tags)
401	Unauthorized
403	Forbidden — ROLE_MANAGER required
404	Template not found

Add or Update Template Tags (Concurrent-Safe)

Endpoint: POST /api/v1/templates/{id}/tags

Path Parameters:

- id: UUID of the template

Authorization: Requires ROLE_MANAGER authority

Request Body:

```
{
  "tags": [
    { "key": "campaign", "value": "Q3-2026" }
  ]
}
```

Tag Validation Rules: Same as PUT.

Behavior: Additive upsert — for each tag, if a tag with the same key already exists, its value is updated; otherwise a new tag is created. Existing tags with other keys are not affected. This is the preferred method for interactive clients to avoid lost-update race conditions.

Response: ApiResponseDto<List<Tag>> — returns **all** tags on the template after the operation.

Error Responses:

HTTP Status	Description
400	Validation error (blank key, invalid characters, too many tags)
401	Unauthorized
403	Forbidden — ROLE_MANAGER required
404	Template not found

Delete All Template Tags

Endpoint: DELETE /api/v1/templates/{id}/tags

Path Parameters:

- id: UUID of the template

Authorization: Requires ROLE_MANAGER authority

Response: ApiResponseDto<Void>

HTTP Status: 204 No Content

```

{
  "meta": {
    "timestamp": "2026-03-28T12:00:00Z",
    "correlationId": "tags-del-456",
    "requestId": "req-tagdel-001",
    "status": "success"
  },
  "data": null,
  "errors": []
}

```

Error Responses:

HTTP Status	Description
401	Unauthorized
403	Forbidden — ROLE_MANAGER required
404	Template not found

Delete Template Tag by Key

Endpoint: DELETE /api/v1/templates/{id}/tags/{key}

Path Parameters:

- id: UUID of the template
- key: Tag key to remove (e.g. phase)

Authorization: Requires ROLE_MANAGER authority

Behavior: Removes the tag with the specified key. Idempotent — if the key does not exist, the operation succeeds silently.

Response: ApiResponseDto<Void>

HTTP Status: 204 No Content

Error Responses:

HTTP Status	Description
401	Unauthorized
403	Forbidden — ROLE_MANAGER required
404	Template not found

Parameter Types and Validation

Supported Parameter Types

The service accepts parameter values in their native JSON types for optimal type safety and developer experience:

Type	Description	Native JSON Example	String Alternative	Validation Rules
String	Text value	"Customer Name"	"Customer Name"	Max length: 1000 characters
Integer	Whole number	42	"42"	Range: -2,147,483,648 to 2,147,483,647
Long	Large whole number	9223372036854775807	"9223372036854775807"	Passed as-is to JasperReports
Double	Decimal number	123.45	"123.45"	Passed as-is to JasperReports
Float	Decimal number	123.45	"123.45"	Passed as-is to JasperReports
BigDecimal	Precise decimal	"123.45"	"123.45"	Converted to BigDecimal
Boolean	True/false value	true	"true"	Must be boolean or "true"/"false"
Date	Date value	"2025-10-10"	"2025-10-10"	ISO 8601 format (YYYY-MM-DD)
LocalDate	Local date	"2025-10-10"	"2025-10-10"	Treated as Date
LocalDateTime	Local date/time	"2025-10-10T14:30:00"	"2025-10-10T14:30:00"	Treated as DateTime
Timestamp	SQL timestamp	"2025-10-10T14:30:00Z"	"2025-10-10T14:30:00Z"	Treated as DateTime
Collection	Collection of objects	[{"key": "value"}]	Not supported	Passed as-is to JasperReports
List	List of objects	[{"key": "value"}]	Not supported	Passed as-is to JasperReports
Map	Key-value pairs	{"nested": {"data": "value"}}	Not supported	Passed as-is to JasperReports

Note: Types marked "Passed as-is to JasperReports" are not explicitly validated by the service - the value is passed directly to the report engine for processing.

Type Safety Benefits:

- Numeric values (42, 123.45) are automatically validated and converted
- Boolean values (true/false) provide type safety without string parsing
- Backward compatibility maintained - string values still accepted and converted
- Improved IDE support and API documentation clarity
- **Optional type override:** type field gives callers explicit control over the Java type (e.g. force Integer vs Double)
- **Optional format pattern:** format field controls output shape of numbers and dates in DOCX templates (e.g. #, ##0.00, dd.MM.yyyy)

User Management Contract

User List Contract

Endpoint: GET /api/v1/users

Authorization: Requires ROLE_ADMIN

Query Parameters:

- page (optional): Page number for pagination (default: 1, 1-based)
- size (optional): Page size (default: 20, max: 100)
- search (optional): Search term to filter users by username, email, first name, or last name
- username (optional): Filter by username (specific field filter, case-insensitive)
- email (optional): Filter by email address (specific field filter, case-insensitive)
- firstName (optional): Filter by first name (specific field filter, case-insensitive)
- lastName (optional): Filter by last name (specific field filter, case-insensitive)
- roles (optional): Filter by roles (comma-separated, e.g., ROLE_ADMIN, ROLE_USER)
- enabled (optional): Filter by enabled status (true/false)
- isOAuthUser (optional): Filter by OAuth user status (true/false)
- createdAfter (optional): Filter users created after this date (ISO 8601 format)
- createdBefore (optional): Filter users created before this date (ISO 8601 format)
- lastLoginAfter (optional): Filter users who logged in after this date (ISO 8601 format)
- lastLoginBefore (optional): Filter users who logged in before this date (ISO 8601 format)
- sortBy (optional): Field to sort by (username, email, firstName, lastName, created, lastLogin). Default: username
- sortDir (optional): Sort direction (asc, desc). Default: asc
- exactMatch (optional): If true, all text filters (search, username, email, firstName, lastName) use exact matching instead of partial LIKE matching. Default: false

Response: ApiResponseDto<PaginatedUsersDto>

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "users-list-123",
    "requestId": "req-users-001",
    "status": "success"
  },
  "data": {
    "totalItems": 25,
    "totalPages": 2,
    "currentPage": 1,
    "itemsPerPage": 20,
    "items": [
      {
        "id": "550e8400-e29b-41d4-a716-446655440000",
        "username": "john.doe",
        "email": "john.doe@example.com",
        "firstName": "John",
        "lastName": "Doe",
        "roles": ["ROLE_USER", "ROLE_MANAGER"],
        "enabled": true,

```

```

        "isOAuthUser": false,
        "created": "2025-01-01T00:00:00Z",
        "lastLogin": "2025-10-10T08:15:00Z"
    }
  ]
},
"errors": []
}

```

Current User Profile Contract

Endpoint: GET /api/v1/users/me

Response:

```

{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "user-profile-456",
    "requestId": "req-profile-001",
    "status": "success"
  },
  "data": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "username": "integration_user",
    "email": "integration@external-org.com",
    "firstName": "Integration",
    "lastName": "User",
    "roles": ["ROLE_USER", "ROLE_API_ACCESS"],
    "enabled": true,
    "isOAuthUser": false,
    "created": "2025-09-15T09:00:00Z",
    "lastLogin": "2025-10-10T08:15:00Z"
  },
  "errors": []
}

```

Change User Password Contract

Endpoint: PUT /api/v1/users/{userId}/password (Admin or own password)

Path Parameters:

- userId: UUID of the user whose password to change

Authorization: Admin can change any user's password; users can change their own password

Request Body: ChangePasswordRequestDto

```

{
  "currentPassword": "currentpassword123",
  "newPassword": "newpassword456"
}

```

Response: ApiResponseDto<MessageDto> (HTTP 200 OK)

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "pwd-change-123-abc",
    "requestId": "req-pwd-001",
    "status": "success"
  },
  "data": {
    "message": "Password changed successfully"
  },
  "errors": []
}
```

Update User Roles Contract

Endpoint: PUT /api/v1/users/{userId}/roles (Admin only)

Path Parameters:

- userId: UUID of the user whose roles to update

Authorization: Requires ROLE_ADMIN

Request Body: UpdateRolesRequestDto

```
{
  "roles": ["ROLE_USER", "ROLE_MANAGER"]
}
```

Response: ApiResponseDto<UserDto>

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "role-update-123-abc",
    "requestId": "req-role-001",
    "status": "success"
  },
  "data": {
    "id": "550e8400-e29b-41d4-a716-446655440000",
    "username": "john.doe",
    "email": "john.doe@example.com",
    "firstName": "John",
    "lastName": "Doe",
    "roles": ["ROLE_USER", "ROLE_MANAGER"],
    "created": "2025-01-01T00:00:00Z",
    "lastLogin": "2025-10-10T14:25:00Z"
  },
  "errors": []
}
```

Configuration Management Contract

Get All Configuration Contract

Endpoint: GET /api/v1/config (Admin only)

Authorization: Requires ROLE_ADMIN

Description: Retrieve all system configuration settings including application metadata, JasperReports configuration, messaging settings, audit configuration, system information, and feature flags.

Response: ApiResponseDto<Map<String, Object>>

```
{
  "meta": {
    "timestamp": "2026-02-16T15:51:13Z",
    "correlationId": "779480a0-e152-4ab0-9390-1398e9842fb9",
    "requestId": "ff01ee9f29ea78f11f0e0778e67add3d",
    "status": "success"
  },
  "data": {
    "application": {
      "name": "document-generation-service",
      "version": "1.9.0",
      "configVersion": "2.0",
      "activeProfiles": "production",
      "serverPort": 8080,
      "lastModified": "2026-02-16T15:51:13.660001832"
    },
    "jasperReports": {
      "documentDirParameterName": "REPORTS_DIR",
      "configPath": "/config/jasperreports.properties",
      "failOnMissingConfig": false
    },
    "storage": {
      "uploadDir": "/app/uploads/templates/",
      "tempDir": "/app/temp/cache/",
      "outputDir": "/app/temp/output/"
    },
    "document": {
      "cleanupDelaySeconds": 5,
      "cleanupEnabled": true
    },
    "authentication": {
      "authenticationDisabled": false,
      "internalEnabled": true,
      "oauth2ClientEnabled": true,
      "oauth2WebEnabled": false
    },
    "messaging": {
      "enabled": true,
      "brokerUrl": "tcp://localhost:61616",
      "requestQueue": "document.generation.requests",
      "deadLetterQueue": "document.generation.dlq",
      "workerThreads": 10,
      "maxRetries": 3,
      "processingTimeoutSeconds": 300,
      "maxDocumentSizeMb": 10,

```

```

    "maxQueueSize": 1000,
    "cleanup": {
      "enabled": true,
      "retentionHours": 168,
      "cron": "0 0 3 * * ?"
    }
  },
  "audit": {
    "asyncEnabled": true,
    "detailedEnabled": true,
    "threadPool": {
      "corePoolSize": 2,
      "maxPoolSize": 10,
      "queueCapacity": 1000
    },
    "retention": {
      "standardRetentionDays": 365,
      "notableEventsRetentionDays": 1825
    },
    "cleanup": {
      "enabled": true,
      "batchSize": 1000,
      "preserveCriticalEvents": true
    }
  },
  "system": {
    "javaVersion": "17.0.18",
    "javaVendor": "Eclipse Adoptium",
    "osName": "Linux",
    "osVersion": "5.10.134-013.4.kangaroo.al8.x86_64",
    "osArch": "amd64",
    "availableProcessors": 4,
    "maxMemory": 1610612736,
    "totalMemory": 1610612736,
    "freeMemory": 1474061824,
    "usedMemory": 136550912,
    "uptimeMillis": 3600000
  },
  "features": {
    "auditingEnabled": true,
    "cachingEnabled": true,
    "asyncGenerationEnabled": true,
    "debugMode": false
  },
  "logging": {
    "rootLevel": "INFO"
  }
},
"errors": []
}

```

Configuration Sections

Section	Description
application	Application metadata including name, version, active profiles, and server port
jasperReports	JasperReports engine configuration (template directory parameter, external config path)

Section	Description
storage	File storage paths for uploads, temporary files, and output
document	Document cleanup settings (delay, enabled state)
authentication	Authentication provider settings (internal, OAuth2 client, OAuth2 web)
messaging	Async document generation queue settings (ActiveMQ configuration, worker threads, retries)
audit	Audit system settings including thread pool, retention periods, and cleanup configuration
system	Runtime system information (Java version, OS, memory, CPU, uptime)
features	Feature flags indicating enabled/disabled capabilities
logging	Current logging configuration

Audit and Monitoring Contract

Audit Log Query Contract

Endpoint: GET /api/v1/audit/logs

Query Parameters:

- startDate: Start date for log filtering (ISO 8601)
- endDate: End date for log filtering (ISO 8601)
- eventType: Filter by event type
- userId: Filter by specific user ID
- correlationId: Filter by correlation ID for request tracing
- page: Page number (default: 1, 1-based)
- size: Page size (default: 20, max: 100)

Response:

```
{
  "meta": {
    "timestamp": "2025-10-10T12:30:00Z",
    "correlationId": "audit-query-789",
    "requestId": "req-audit-001",
    "status": "success"
  },
  "data": {
    "auditLogs": [
      {
        "id": "audit-550e8400-e29b-41d4-a716-446655440000",
        "timestamp": "2025-10-10T12:25:00Z",
        "eventType": "TEMPLATE_GENERATED",
        "userId": "550e8400-e29b-41d4-a716-446655440000",
        "username": "integration_user",
        "correlationId": "gen-abc-123",
        "resourceType": "TEMPLATE",
        "resourceId": "550e8400-e29b-41d4-a716-446655440000",
        "action": "Generate PDF document",
        "description": "Successfully generated PDF document from invoice template",
        "executionTimeMs": 1247,
        "responseStatus": 200,
        "severity": "LOW",
        "ipAddress": "192.168.1.100",
        "userAgent": "IntegrationClient/1.0"
      }
    ],
    "pagination": {
      "page": 1,
      "size": 20,
      "totalElements": 150,
      "totalPages": 8,
      "first": true,
      "last": false
    }
  },
  "errors": []
}
```

Audit Statistics Contract

Endpoint: GET /api/v1/audit/statistics (Admin only)

Authorization: Requires ROLE_ADMIN

Query Parameters:

- startTime: Start time for statistics (ISO 8601, optional)
- endTime: End time for statistics (ISO 8601, optional)

Response: ApiResponseDto<AuditStatisticsDto>

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "audit-stats-123-abc",
    "requestId": "req-stats-001",
    "status": "success"
  },
  "data": {
    "totalEvents": 1250,
    "successfulEvents": 1180,
    "failedEvents": 70,
    "eventsByType": {
      "TEMPLATE_GENERATED": 450,
      "USER_CREATED": 25,
      "LOGIN_SUCCESS": 320,
      "LOGIN_FAILURE": 15
    },
    "eventsBySeverity": {
      "LOW": 1000,
      "MEDIUM": 200,
      "HIGH": 35,
      "CRITICAL": 15
    },
    "eventsByUser": [
      {
        "userId": "550e8400-e29b-41d4-a716-446655440000",
        "username": "admin@example.com",
        "eventCount": 450
      },
      {
        "userId": "660e8400-e29b-41d4-a716-446655440001",
        "username": "user1@example.com",
        "eventCount": 320
      }
    ],
    "eventsByClient": [
      {
        "clientId": "mobile-app-client",
        "eventCount": 580
      },
      {
        "clientId": "web-portal-client",
        "eventCount": 420
      }
    ],
    "eventsByDay": {
```

```

    "2025-10-01": 125,
    "2025-10-02": 138,
    "2025-10-03": 142
  },
  "topIpAddresses": [
    {
      "ipAddress": "192.168.1.100",
      "eventCount": 250
    },
    {
      "ipAddress": "10.0.0.50",
      "eventCount": 180
    }
  ],
  "uniqueUsers": 45,
  "uniqueIpAddresses": 78,
  "averageExecutionTime": 245.5
},
"errors": []
}

```

AuditStatisticsDto Fields:

- `totalEvents`: Total number of audit events in the period
- `successfulEvents`: Count of successful operations
- `failedEvents`: Count of failed operations
- `eventsByType`: Map of event type to count (TEMPLATE_GENERATED, LOGIN_SUCCESS, etc.)
- `eventsBySeverity`: Map of severity level to count (LOW, MEDIUM, HIGH, CRITICAL)
- `eventsByUser`: Top 20 most active users with `userId`, `username`, and event counts
- `eventsByClient`: Top 20 most active OAuth2 clients with `clientId` and event counts (includes client authentication)
- `eventsByDay`: Daily event counts mapped by date (ISO date format)
- `topIpAddresses`: Top 10 IP addresses by activity with event counts
- `uniqueUsers`: Total count of unique users who generated events
- `uniqueIpAddresses`: Total count of unique IP addresses
- `averageExecutionTime`: Average execution time in milliseconds across all events

User-Specific Audit Logs Contract

Endpoint: GET `/api/v1/audit/users/{userId}` (Admin only)

Authorization: Requires `ROLE_ADMIN`

Path Parameters:

- `userId`: UUID of the user to get audit logs for

Query Parameters:

- `page`: Page number (default: 1)
- `size`: Page size (default: 50, max: 100)
- `startTime`: Start time for filtering (ISO 8601, optional)
- `endTime`: End time for filtering (ISO 8601, optional)

Response: `ApiResponseDto<PaginatedAuditLogsDto>`

Security Events Contract

Endpoint: GET /api/v1/audit/security (Admin only)

Authorization: Requires ROLE_ADMIN

Description: Get high-priority security-related audit logs including failed logins, unauthorized access attempts, and suspicious activities.

Query Parameters:

- page: Page number (default: 1)
- size: Page size (default: 50, max: 100)
- startTime: Start time for filtering (ISO 8601, optional)
- endTime: End time for filtering (ISO 8601, optional)

Response: ApiResponseDto<PaginatedAuditLogsDto>

Audit System Health Contract

Endpoint: GET /api/v1/audit/health

Description: Health check endpoint for audit system.

Response: ApiResponseDto<String>

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "audit-health-123-abc",
    "requestId": "req-health-001",
    "status": "success"
  },
  "data": "Audit system is operational",
  "errors": []
}
```

Manual Audit Cleanup Contract

Endpoint: POST /api/v1/audit/cleanup (Admin only)

Authorization: Requires ROLE_ADMIN

Description: Trigger manual cleanup of old audit logs.

Response: ApiResponseDto<String>

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "audit-cleanup-123-abc",
    "requestId": "req-cleanup-001",
    "status": "success"
  },
  "data": "Audit log cleanup initiated",
  "errors": []
}
```

Audit Dashboard Overview Contract

Endpoint: GET /api/v1/audit/dashboard/overview (Admin only)

Authorization: Requires ROLE_ADMIN

Description: Get real-time audit dashboard overview with activity statistics and recent critical events.

Response: ApiResponseDto<Map<String, Object>>

```
{
  "meta": {
    "timestamp": "2025-10-10T14:30:00Z",
    "correlationId": "dashboard-overview-123-abc",
    "requestId": "req-dashboard-001",
    "status": "success"
  },
  "data": {
    "last24Hours": {
      "totalEvents": 145,
      "successfulEvents": 140,
      "failedEvents": 5
    },
    "last7Days": {
      "totalEvents": 890,
      "successfulEvents": 860,
      "failedEvents": 30
    },
    "recentCriticalEvents": [],
    "recentFailures": [],
    "recentSecurityEvents": []
  },
  "errors": []
}
```

Security Threats Dashboard Contract

Endpoint: GET /api/v1/audit/dashboard/security-threats (Admin only)

Authorization: Requires ROLE_ADMIN

Description: Get security threat summary for the last 24 hours including failed logins, unauthorized access attempts, and suspicious activities.

Response: ApiResponseDto<Map<String, Object>>

```
{
  "meta": {
    "timestamp": "2025-10-23T20:00:00Z",
    "correlationId": "security-threats-123-abc",
    "requestId": "req-security-001",
    "status": "success"
  },
  "data": {
    "threatLevel": "MEDIUM",
    "failedLogins": 12,
    "unauthorizedAccess": 3,
    "suspiciousActivities": 7,
    "lastThreatDetected": "2025-10-23T19:45:00Z",
  }
}
```

```
    "criticalAlerts": 2
  },
  "errors": []
}
```

Compliance Dashboard Contract

Endpoint: GET /api/v1/audit/dashboard/compliance (Admin only)

Authorization: Requires ROLE_ADMIN

Query Parameters:

- startTime: Start time for compliance analysis (ISO 8601, optional)
- endTime: End time for compliance analysis (ISO 8601, optional)

Description: Get compliance activity summary including critical operations requiring compliance logging.

Response: ApiResponseDto<Map<String, Object>>

```
{
  "meta": {
    "timestamp": "2025-10-23T20:00:00Z",
    "correlationId": "compliance-456-def",
    "requestId": "req-compliance-001",
    "status": "success"
  },
  "data": {
    "complianceScore": 98.5,
    "criticalOperations": 156,
    "auditTrailComplete": true,
    "lastComplianceCheck": "2025-10-23T19:30:00Z",
    "violations": 0,
    "pendingReviews": 3
  },
  "errors": []
}
```

System Health Metrics Contract

Endpoint: GET /api/v1/audit/dashboard/system-health (Admin only)

Authorization: Requires ROLE_ADMIN

Description: Get system health metrics from audit perspective including error rates and system events.

Response: ApiResponseDto<Map<String, Object>>

```
{
  "meta": {
    "timestamp": "2025-10-23T20:00:00Z",
    "correlationId": "system-health-789-ghi",
    "requestId": "req-health-002",
    "status": "success"
  },
  "data": {
    "systemHealth": "HEALTHY",
    "errorRate": 0.02,
  }
}
```

```
"averageResponseTime": 245,  
"systemEvents": 1847,  
"lastSystemCheck": "2025-10-23T19:55:00Z",  
"warningThreshold": 0.05,  
"criticalThreshold": 0.10  
},  
"errors": []  
}
```

File Integrity Management Contract

Overview

The File Integrity Management API provides cryptographic verification of template files using HMAC-SHA256 digests. This ensures that template files have not been tampered with or corrupted. All file integrity endpoints require ROLE_ADMIN authorization.

Regenerate Digest Contract

Endpoint: POST /api/v1/admin/templates/{id}/regenerate-digest

Path Parameters:

- id: UUID of the template document

Authorization: Requires ROLE_ADMIN

Description: Regenerates the HMAC-SHA256 digest for a specific template file. Use this after secret key rotation or if the digest file is missing or corrupted.

Audit Event: This operation automatically triggers an INTEGRITY_DIGEST_REGENERATED audit event with MEDIUM severity, including:

- Resource Type: DOCUMENT
- Resource ID: Template UUID
- User ID and IP address
- Timestamp and correlation ID

Response: ApiResponseDto<String>

HTTP Status: 200 OK

```
{
  "meta": {
    "timestamp": "2025-11-25T12:30:00Z",
    "correlationId": "integrity-regen-123-abc",
    "requestId": "req-integrity-001",
    "status": "success"
  },
  "data": "Digest regenerated successfully for document: 550e8400-e29b-41d4-a716-446655440000",
  "errors": []
}
```

Verify Integrity Contract

Endpoint: POST /api/v1/admin/templates/{id}/verify-integrity

Path Parameters:

- id: UUID of the template document

Authorization: Requires ROLE_ADMIN

Description: Verifies the integrity of a template file by recalculating its HMAC-SHA256 digest and comparing it with the stored digest.

Audit Events: This operation triggers multiple audit events:

- INTEGRITY_CHECK_PERFORMED (LOW severity) - When verification is initiated
- INTEGRITY_CHECK_PASSED (LOW severity) - When verification succeeds
- INTEGRITY_CHECK_FAILED [!] **CRITICAL** (HIGH severity) - When verification fails, indicating possible file tampering

Failed integrity checks are classified as critical security events and trigger immediate security monitoring alerts.

Response: ApiResponseDto<String>

HTTP Status: 200 OK (integrity verified) or 409 Conflict (integrity violation)

Success Response:

```
{
  "meta": {
    "timestamp": "2025-11-25T12:31:00Z",
    "correlationId": "integrity-verify-456-def",
    "requestId": "req-integrity-002",
    "status": "success"
  },
  "data": "File integrity verified successfully for document: 550e8400-e29b-41d4-a716-446655440000",
  "errors": []
}
```

Integrity Violation Response (HTTP 409):

```
{
  "meta": {
    "timestamp": "2025-11-25T12:32:00Z",
    "correlationId": "integrity-violation-789-ghi",
    "requestId": "req-integrity-003",
    "status": "error"
  },
  "data": null,
  "errors": [
    {
      "code": "INTEGRITY_VIOLATION",
      "message": "Template file integrity check failed for document:
        ↪ 550e8400-e29b-41d4-a716-446655440000",
      "details": "The stored digest does not match the calculated digest. File may have been tampered
        ↪ with or corrupted."
    }
  ]
}
```

Regenerate All Digests Contract

Endpoint: POST /api/v1/admin/templates/regenerate-all-digests

Authorization: Requires ROLE_ADMIN

Description: Regenerates HMAC-SHA256 digests for all template files in the system. Use this after rotating the secret key used for digest calculation.

Audit Event: This bulk operation triggers an INTEGRITY_BULK_REGENERATION audit event with HIGH severity, documenting the administrative action across all templates.

Response: ApiResponseDto<String>

HTTP Status: 200 OK

```
{
  "meta": {
    "timestamp": "2025-11-25T12:33:00Z",
    "correlationId": "integrity-bulk-123-xyz",
    "requestId": "req-integrity-004",
    "status": "success"
  },
  "data": "Successfully regenerated digests for 45 templates",
  "errors": []
}
```

Common Use Cases

Secret Key Rotation

1. Update the secret key in configuration:

```
file.integrity.secret-key=new-secret-key-value
```

2. Regenerate all digests:

```
curl -X POST http://localhost:8080/api/v1/admin/templates/regenerate-all-digests \
  -H "Authorization: Bearer <admin-token>"
```

Security Audit

Verify integrity of all critical templates:

```
# Get List of templates
curl -X GET http://localhost:8080/api/v1/templates \
  -H "Authorization: Bearer <admin-token>"

# Verify each template
for template_id in $template_ids; do
  curl -X POST http://localhost:8080/api/v1/admin/templates/$template_id/verify-integrity \
    -H "Authorization: Bearer <admin-token>"
done
```

Missing Digest Recovery

If a .integrity file is missing or deleted:

```
curl -X POST http://localhost:8080/api/v1/admin/templates/{id}/regenerate-digest \
  -H "Authorization: Bearer <admin-token>"
```

File Integrity Security

1. **Secret Key Security:** The HMAC secret key should be stored securely in environment variables or secret management systems
2. **Key Rotation:** Regular key rotation is recommended for enhanced security
3. **Comprehensive Audit Logging:** All integrity operations are automatically logged with dedicated audit event types:
 - INTEGRITY_DIGEST_REGENERATED - Single template digest regeneration
 - INTEGRITY_CHECK_PERFORMED - Integrity verification attempt

- INTEGRITY_CHECK_PASSED - Successful verification
 - INTEGRITY_CHECK_FAILED - Failed verification (critical security event)
 - INTEGRITY_BULK_REGENERATION - Bulk digest regeneration
4. **Access Control:** Only administrators with ROLE_ADMIN can perform integrity operations
 5. **Digest Storage:** Digest files are stored alongside template files with `.integrity` extension
 6. **Real-time Monitoring:** Failed integrity checks trigger immediate security alerts for investigation

For detailed implementation information, see [FILE_INTEGRITY_SYSTEM.md](#).

Document Post-Processing Contract

The post-processing API allows applying PDF transformations to pre-generated documents without re-rendering from a template. This is useful when both a plain (archival) and a secured (delivery) version of the same document are needed.

PDF Post-Processing

Endpoint: POST /api/v1/documents/post-process/pdf

Content-Type: multipart/form-data

Authorization: Requires authentication (any authenticated user)

Form Parts:

Part	Type	Required	Description
file	Binary (PDF)	Yes	The PDF file to post-process. Must start with %PDF magic bytes
options	JSON	Yes	Post-processing options — same schema as pdfExportOptions in the generation endpoint

Query Parameters:

Parameter	Type	Required	Default	Description
filename	String	No	Original filename	Desired filename for the response (.pdf extension ensured)

Options Schema (PdfExportOptionsDto):

The options JSON part accepts the full pdfExportOptions schema documented in the Document Generation Contract. Key fields:

- fontEmbeddingSubstitute — font family to replace non-embedded base-14 fonts
- imageCompressionQuality — JPEG quality (0.0–1.0) for image re-compression
- cmykConversionProfile — ICC profile for RGB→CMYK conversion
- cmykPreserveBlack — Preserve Black during RGB→CMYK (default true)
- flattenTransparency — remove redundant transparency groups
- pdfaConformance — PDF/A conformance level (e.g., "PDF/A-1b")
- iccProfile — ICC profile for PDF/A OutputIntent
- duplexPadding — pad to even page count for duplex printing
- userPassword, ownerPassword — PDF encryption passwords
- canPrint, canCopy, canModify, etc. — access permissions
- encryptionKeyLength — 128 or 256 bits

Important: PDF/A conformance and encryption are mutually exclusive (ISO 19005). When both are specified, PDF/A is applied and security is skipped with a warning.

Processing Pipeline:

Operations are applied sequentially in this order:

1. Font embedding
2. Image compression
3. CMYK colour conversion
4. Transparency flattening
5. PDF/A conformance
6. Duplex padding
7. Security (encryption + permissions)

This is the same pipeline used during template-based PDF generation.

Example Request:

```
curl -X POST "http://localhost:8080/api/v1/documents/post-process/pdf" \
  -H "Authorization: Bearer <token>" \
  -F "file=@plain_report.pdf" \
  -F 'op-
  ↪ tions={"userPassword":"viewer123","ownerPassword":"admin456","canPrint":true,"canCopy":false,"encryptionKeyLen
  ↪ \
  -G -d "filename=secured_report"
```

Example — PDF/A Conversion:

```
curl -X POST "http://localhost:8080/api/v1/documents/post-process/pdf" \
  -H "Authorization: Bearer <token>" \
  -F "file=@document.pdf" \
  -F 'options={"pdfaConformance":"PDF/A-1b","iccProfile":"pso-coated-v3"};type=application/json'
```

Example — Image Optimisation + CMYK:

```
curl -X POST "http://localhost:8080/api/v1/documents/post-process/pdf" \
  -H "Authorization: Bearer <token>" \
  -F "file=@brochure.pdf" \
  -F 'options={"imageCompressionQuality":0.85,"cmykConversionProfile":"coated-
  ↪ fogra39"};type=application/json'
```

Successful Response:

```
HTTP/1.1 200 OK
Content-Type: application/pdf
Content-Disposition: attachment; filename="secured_report.pdf"
X-Processing-Time-Ms: 842
```

[Post-processed PDF binary data]

Error Responses:

HTTP Status	Error Code	Description
400	VALIDATION_ERROR	Uploaded file is not a valid PDF (missing %PDF header)
400	VALIDATION_ERROR	Invalid options JSON
401	AUTHENTICATION_FAILED	Missing or invalid authentication token
413	—	File exceeds maximum upload size
500	INTERNAL_ERROR	Post-processing pipeline failure

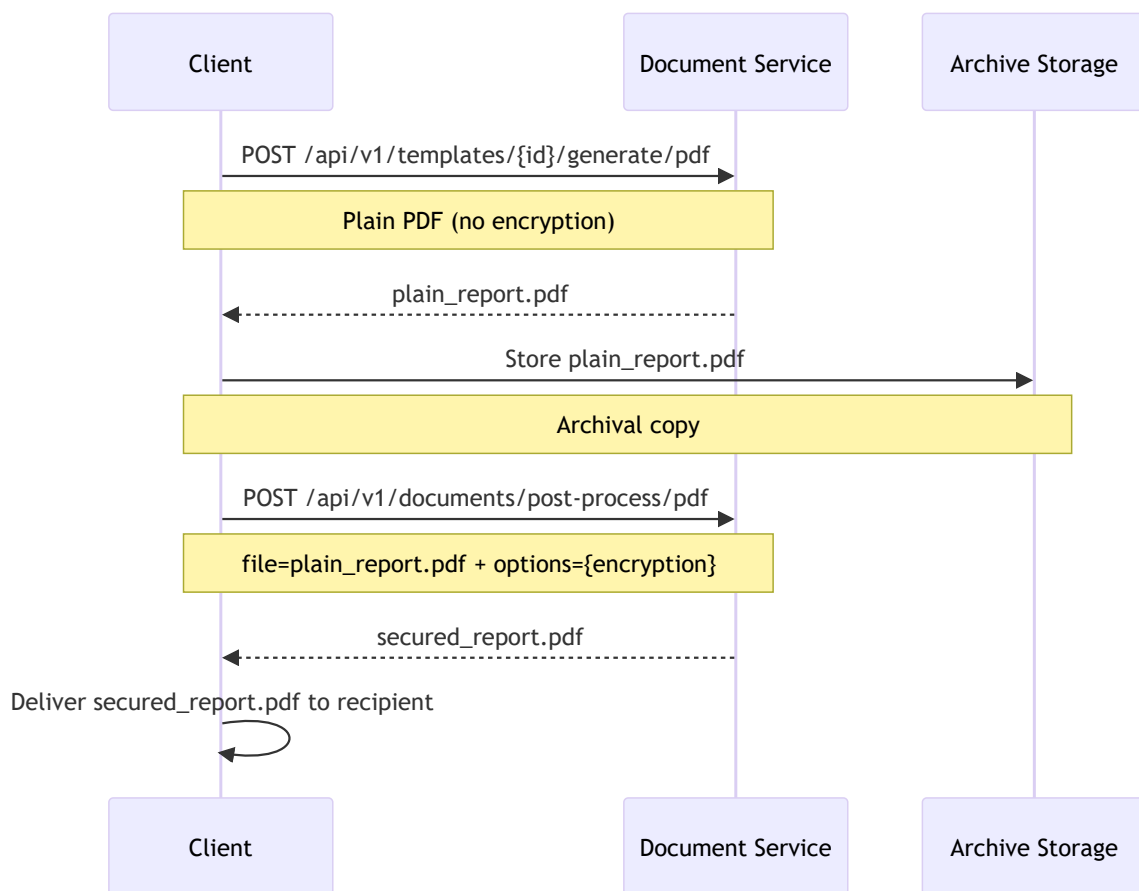
Validation Error Example:

```

{
  "meta": {
    "timestamp": "2026-03-31T10:00:00Z",
    "correlationId": "pp-error-123",
    "requestId": "req-pp-001",
    "status": "error"
  },
  "data": null,
  "errors": [
    {
      "code": "VALIDATION_ERROR",
      "message": "The uploaded file is not a valid PDF document"
    }
  ]
}

```

Typical Use Case — Archive + Secure Delivery:



Audit:

All post-processing requests are logged with event type DOCUMENT_POSTPROCESSED in the audit system.

Async Monitoring API (Optional)

The async monitoring API provides REST endpoints for monitoring asynchronous document generation when enabled (`document.async.enabled=true`). All endpoints require `ROLE_ADMIN` authorization.

Async Metrics Dashboard Contract

Endpoint: GET `/api/v1/async/metrics`

Description: Returns comprehensive dashboard metrics including queue depth, performance statistics, throughput, and error rates for async document generation.

Authorization: Requires `ROLE_ADMIN`

Response: `ApiResponseDto<AsyncMetricsDto>`

```
{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "metrics-abc-123",
    "requestId": "req-metrics-001",
    "status": "success"
  },
  "data": {
    "queueDepth": {
      "queued": 5,
      "processing": 2,
      "completed": 1500,
      "failed": 12,
      "timeout": 1
    },
    "performance": {
      "avgProcessingTimeMs": 2340.5,
      "last24h": 2450.2,
      "last1h": 2120.8
    },
    "throughput": {
      "last24h": 1500,
      "last1h": 125,
      "last15min": 32
    },
    "errorRate": {
      "last24h": 0.8,
      "last1h": 0.5
    }
  },
  "errors": []
}
```

Response Fields:

- `queueDepth`: Current count of requests by status
 - `queued`: Requests waiting in queue
 - `processing`: Requests currently being processed
 - `completed`: Successfully completed requests
 - `failed`: Failed requests
 - `timeout`: Timed-out requests

- performance: Processing time statistics (in milliseconds)
 - avgProcessingTimeMs: Overall average processing time
 - last24h: Average processing time in last 24 hours
 - last1h: Average processing time in last hour
- throughput: Number of completed requests
 - last24h, last1h, last15min: Completed requests in respective time periods
- errorRate: Failure percentage
 - last24h, last1h: Error rate as percentage (0-100)

Async Request List Contract

Endpoint: GET /api/v1/async/requests

Description: Returns a paginated list of async requests with optional filtering by status, user, template, and time range.

Authorization: Requires ROLE_ADMIN

Query Parameters:

Parameter	Type	Required	Default	Description
status	String	No	-	Filter by status: QUEUED, PROCESSING, COMPLETED, FAILED, TIMEOUT
userId	String	No	-	Filter by user ID
templateId	UUID	No	-	Filter by template ID
since	DateTime	No	-	Filter requests created after this timestamp (ISO 8601 format)
page	Integer	No	1	Page number (1-based)
size	Integer	No	20	Items per page (max: 100)

Response: ApiResponseDto<PaginatedAsyncRequestsDto>

```
{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "requests-abc-123",
    "requestId": "req-list-001",
    "status": "success"
  },
  "data": {
    "totalItems": 150,
    "totalPages": 8,
    "currentPage": 1,
    "itemsPerPage": 20,
    "items": [
      {
        "requestId": "550e8400-e29b-41d4-a716-446655440000",
        "correlationId": "client-correlation-123",
        "templateId": "template-uuid-here",
        "format": "PDF",
        "userId": "user@example.com",
        "status": "COMPLETED",
        "priority": 5,

```

```

    "createdAt": "2025-12-30T09:55:00Z",
    "startedAt": "2025-12-30T09:55:01Z",
    "completedAt": "2025-12-30T09:55:03Z",
    "elapsedMs": 2340,
    "retryCount": 0
  }
]
},
"errors": []
}

```

Response Fields:

- totalItems: Total number of matching requests
- totalPages: Total number of pages
- currentPage: Current page number (1-based)
- itemsPerPage: Number of items per page
- items: Array of request summaries
 - requestId: Unique request identifier (UUID)
 - correlationId: Client-provided correlation ID
 - templateId: Template used for generation
 - format: Output format (PDF, XLSX, DOCX, TXT, etc.)
 - userId: User who submitted the request
 - status: Current status (QUEUED, PROCESSING, COMPLETED, FAILED, TIMEOUT)
 - priority: Request priority (1-10, default: 5)
 - createdAt: Request creation timestamp
 - startedAt: Processing start timestamp (null if queued)
 - completedAt: Completion timestamp (null if not completed)
 - elapsedMs: Total elapsed time in milliseconds
 - retryCount: Number of retry attempts

Async Request Details Contract

Endpoint: GET /api/v1/async/requests/{requestId}

Description: Returns detailed information about a specific async request including full metrics and error information for troubleshooting.

Authorization: Requires ROLE_ADMIN

Path Parameters:

- requestId (required): UUID of the async request

Response: ApiResponseDto<AsyncRequestDetailsDto>

HTTP Status: 200 OK if found, 404 Not Found if request doesn't exist

```

{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "details-abc-123",
    "requestId": "req-details-001",
    "status": "success"
  }
}

```

```

},
"data": {
  "requestId": "550e8400-e29b-41d4-a716-446655440000",
  "correlationId": "client-correlation-123",
  "templateId": "template-uuid-here",
  "format": "PDF",
  "userId": "user@example.com",
  "status": "COMPLETED",
  "priority": 5,
  "replyQueue": "document.generation.replies.client-abc-123",
  "createdAt": "2025-12-30T09:55:00Z",
  "startedAt": "2025-12-30T09:55:01Z",
  "completedAt": "2025-12-30T09:55:03Z",
  "processingTimeMs": 2100,
  "queueWaitTimeMs": 240,
  "filename": "monthly_report.pdf",
  "fileSize": 245678,
  "base64Size": 327570,
  "retryCount": 0,
  "errorMessage": null
},
"errors": []
}

```

Response Fields:

- All fields from request list, plus:
- replyQueue: JMS reply queue for the client
- processingTimeMs: Time spent generating the document
- queueWaitTimeMs: Time spent waiting in queue
- filename: Generated document filename
- fileSize: Original document size in bytes
- base64Size: Base64-encoded document size in bytes
- errorMessage: Error details (for failed/timeout requests)

Async Worker Status Contract

Endpoint: GET /api/v1/async/workers

Description: Returns JMS listener container status and list of currently processing requests. Useful for monitoring worker health and active operations.

Authorization: Requires ROLE_ADMIN

Response: ApiResponseDto<AsyncWorkersDto>

```

{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "workers-abc-123",
    "requestId": "req-workers-001",
    "status": "success"
  },
  "data": {
    "jmsListenerContainers": [
      {

```

```

    "containerId": "org.springframework.jms.JmsListenerEndpointContainer#0",
    "queueName": "document.generation.requests",
    "concurrency": "1-10",
    "activeSessions": 3,
    "maxSessions": 10,
    "currentlyProcessing": 2,
    "status": "RUNNING"
  }
],
"processingRequests": [
  {
    "requestId": "550e8400-e29b-41d4-a716-446655440000",
    "threadName": "jmsContainer-1",
    "startedAt": "2025-12-30T09:59:58Z",
    "elapsedMs": 2150,
    "templateId": "template-uuid-here",
    "format": "PDF"
  }
]
},
"errors": []
}

```

Response Fields:

- `jmsListenerContainers`: Array of JMS listener container statuses
 - `containerId`: Container identifier
 - `queueName`: Queue being consumed
 - `concurrency`: Concurrency configuration (min-max threads)
 - `activeSessions`: Current active consumer sessions
 - `maxSessions`: Maximum configured sessions
 - `currentlyProcessing`: Number of messages being processed
 - `status`: Container status (RUNNING, STOPPED, PAUSED)
- `processingRequests`: Array of requests currently being processed
 - `requestId`: Request identifier
 - `threadName`: Thread processing the request
 - `startedAt`: Processing start timestamp
 - `elapsedMs`: Time elapsed since processing started
 - `templateId`: Template being used
 - `format`: Output format

Async Error Log Contract

Endpoint: GET `/api/v1/async/errors`

Description: Returns a paginated list of failed or timed-out async requests for troubleshooting and error analysis.

Authorization: Requires `ROLE_ADMIN`

Query Parameters:

Parameter	Type	Required	Default	Description
since	DateTime	No	24 hours ago	Show errors after this timestamp (ISO 8601 format)
page	Integer	No	1	Page number (1-based)
size	Integer	No	20	Items per page (max: 100)

Response: ApiResponseDto<PaginatedAsyncErrorsDto>

```

{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "errors-abc-123",
    "requestId": "req-errors-001",
    "status": "success"
  },
  "data": {
    "totalItems": 12,
    "totalPages": 1,
    "currentPage": 1,
    "itemsPerPage": 20,
    "items": [
      {
        "requestId": "550e8400-e29b-41d4-a716-446655440001",
        "correlationId": "client-correlation-456",
        "templateId": "template-uuid-here",
        "userId": "user@example.com",
        "failedAt": "2025-12-30T09:50:01Z",
        "errorMessage": "Template not found with id: template-uuid-here",
        "retryCount": 3,
        "processingTimeMs": 150
      }
    ]
  },
  "errors": []
}

```

Response Fields:

- totalItems: Total number of errors in the time range
- totalPages: Total number of pages
- currentPage: Current page number (1-based)
- itemsPerPage: Number of items per page
- items: Array of error records
 - requestId: Failed request identifier
 - correlationId: Client-provided correlation ID
 - templateId: Template that was requested
 - userId: User who submitted the request
 - failedAt: Failure timestamp
 - errorMessage: Error description
 - retryCount: Number of retry attempts before failure
 - processingTimeMs: Processing time before failure

Async System Health Contract

Endpoint: GET /api/v1/async/health

Description: Returns Actuator-style health check for async system components including ActiveMQ broker, queue depth, and worker status.

Authorization: Requires ROLE_ADMIN

Response: ApiResponseDto<AsyncHealthDto>

```
{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "health-abc-123",
    "requestId": "req-health-001",
    "status": "success"
  },
  "data": {
    "status": "UP",
    "components": {
      "activemq": {
        "status": "UP",
        "details": {
          "brokerUrl": "tcp://localhost:61616",
          "connectionStatus": "CONNECTED"
        }
      }
    },
    "queueDepth": {
      "status": "UP",
      "details": {
        "queued": 5,
        "processing": 2,
        "thresholdWarning": 100,
        "thresholdCritical": 500
      }
    },
    "workers": {
      "status": "UP",
      "details": {
        "activeContainers": 1,
        "totalActiveSessions": 3,
        "currentlyProcessing": 2
      }
    }
  },
  "errors": []
}
```

Response Fields:

- status: Overall health status (UP, DOWN, DEGRADED)
- components: Individual component health checks
 - Each component has:
 - * status: Component status (UP, DOWN, UNKNOWN)
 - * details: Component-specific details

Health Status Values:

- UP: Component is healthy and operational
- DOWN: Component is not functional
- DEGRADED: Component is operational but with issues (e.g., high queue depth)
- UNKNOWN: Health status cannot be determined

Bulk Async Submission Contract

Endpoint: POST /api/v1/async/bulk

Description: Submit multiple document generation requests in a single HTTP call. This endpoint is optimized for high-volume batch submissions (up to 10,000 requests per call) and provides an alternative to direct JMS queue submission. Requests are validated, queued, and tracked for monitoring.

Authorization: Requires authentication (any authenticated user)

Request Body: BulkAsyncRequestDto

```
{
  "batchCorrelationId": "batch-2025-12-30-001",
  "requests": [
    {
      "requestId": "550e8400-e29b-41d4-a716-446655440000",
      "correlationId": "order-12345",
      "templateId": "123e4567-e89b-12d3-a456-426614174000",
      "format": "PDF",
      "parameters": [
        { "name": "reportTitle", "value": "Monthly Report" },
        { "name": "amount", "value": 1500.50 }
      ],
      "data": {
        "items": [
          { "name": "Item 1", "quantity": 10 }
        ]
      },
      "documentLocale": "pl_PL",
      "pdfExportOptions": {
        "pdfaConformance": "PDF/A-1b"
      },
      "htmlExportOptions": null,
      "txtExportOptions": null,
      "ignorePagination": false,
      "filename": "monthly_report_jan_2025",
      "replyQueue": "document.generation.replies.client-abc",
      "priority": 5,
      "timeoutSeconds": 300
    }
  ]
}
```

Request Fields:

- batchCorrelationId (optional): Correlation ID for the entire batch. If omitted, a UUID is generated.
- requests (required): Array of document generation requests (1 to 10,000 items)
 - requestId (optional): Client-provided UUID. If omitted, a UUID is generated.

- correlationId (optional): Client-provided correlation ID. If omitted, derived as {batchCorrelationId}-{index}.
- templateId (required): UUID of the template to use.
- format (required): Output format (PDF, XLSX, DOCX, RTF, HTML, TXT, CSV, XML, JSON, PPTX).
- parameters (optional): Array of {name, value} objects for JasperReports parameters.
- data (optional): JSON data source for the report.
- documentLocale (optional): Locale for date/number formatting (e.g., en_US, pl_PL).
- pdfExportOptions (optional): PDF-specific export options.
- htmlExportOptions (optional): HTML-specific export options.
- txtExportOptions (optional): TXT-specific export options (characterWidth, characterHeight, pageWidthInChars, pageHeightInChars, lineSeparator, pageSeparator, trimLineRight).
- ignorePagination (optional): Whether to ignore pagination (default: false).
- filename (optional): Output filename without extension.
- replyQueue (optional): JMS reply queue for response delivery.
- priority (optional): Processing priority 1-10 (default: 5).
- timeoutSeconds (optional): Processing timeout in seconds.

Response: ApiResponseDto<BulkAsyncResponseDto>

```

{
  "meta": {
    "timestamp": "2025-12-30T10:00:00Z",
    "correlationId": "bulk-abc-123",
    "requestId": "req-bulk-001",
    "status": "success"
  },
  "data": {
    "batchCorrelationId": "batch-2025-12-30-001",
    "submittedAt": "2025-12-30T10:00:00Z",
    "totalRequests": 1000,
    "successCount": 998,
    "failedCount": 2,
    "queueDepthBefore": 150,
    "estimatedTotalProcessingMs": 300000,
    "queuedRequests": [
      {
        "requestId": "550e8400-e29b-41d4-a716-446655440000",
        "correlationId": "order-12345",
        "index": 0,
        "templateId": "123e4567-e89b-12d3-a456-426614174000",
        "format": "PDF"
      }
    ],
    "failedRequests": [
      {
        "index": 5,
        "correlationId": "order-12350",
        "errorCode": "VALIDATION_ERROR",
        "errorMessage": "Template not found with id: 123e4567-e89b-12d3-a456-426614174999"
      }
    ]
  },
  "errors": []
}

```

Response Fields:

- `batchCorrelationId`: Correlation ID for the batch (client-provided or generated)
- `submittedAt`: Timestamp when batch was queued
- `totalRequests`: Total number of requests submitted
- `successCount`: Number of requests successfully queued
- `failedCount`: Number of requests that failed validation
- `queueDepthBefore`: Queue depth before this batch was added (for estimation)
- `estimatedTotalProcessingMs`: Estimated total processing time based on average processing time and queue depth
- `queuedRequests`: Array of successfully queued requests
 - `requestId`: Generated or client-provided request ID
 - `correlationId`: Generated or client-provided correlation ID
 - `index`: Position in the original request array (0-based)
 - `templateId`: Template UUID
 - `format`: Output format
- `failedRequests`: Array of requests that failed validation (null if none failed)
 - `index`: Position in the original request array (0-based)
 - `correlationId`: Client-provided correlation ID (if any)
 - `errorCode`: Error code (e.g., `VALIDATION_ERROR`)
 - `errorMessage`: Detailed error message

Validation Errors:

Requests are validated individually. Validation failures include:

- Missing or invalid `templateId`
- Template not found
- Missing or unsupported format
- Invalid priority (must be 1-10)

Usage Notes:

- Responses are delivered to the JMS `replyQueue` specified in each request
- Use the `GET /api/v1/async/requests` endpoint to monitor request status
- Large batches are processed efficiently using batch database inserts and JMS publishing
- Failed validation does not affect other requests in the same batch

Async Operations API (Optional)

The Async Operations API provides user-facing endpoints for asynchronous document generation workflow: submitting requests, polling for results, downloading documents, and acknowledging completions. When enabled (`document.async.enabled=true`), users can manage their async requests via HTTP.

Key Features:

- **Dual-Mode Architecture:** Works alongside JMS queue consumption - both patterns can be used simultaneously
- **Non-Destructive Browse:** Status checks use JMS QueueBrowser, leaving messages for JMS clients
- **User Ownership:** Users can only access their own async requests
- **Authorization:** Requires authentication (any authenticated user)

Get Async Result Status Contract

Endpoint: GET `/api/v1/async/results/{requestId}`

Description: Check the status of an async document generation request. For completed requests, optionally includes base64-encoded document content. This is a non-destructive operation - the message remains in the queue for JMS clients.

Authorization: Requires authentication (user can only access own requests)

Path Parameters:

- `requestId` (required): UUID of the async request

Query Parameters:

Parameter	Type	Required	Default	Description
<code>includeContent</code>	Boolean	No	<code>false</code>	Include base64 document content in response

Response: `ApiResponseDto<AsyncResultDto>`

HTTP Status Codes:

- 200 OK: Request completed, result available
- 202 Accepted: Request is still being processed (QUEUED or PROCESSING)
- 404 Not Found: Request not found or not owned by user
- 403 Forbidden: Request belongs to another user
- 503 Service Unavailable: Async processing not enabled or queue unavailable

```
{
  "meta": {
    "timestamp": "2026-01-27T10:00:00Z",
    "correlationId": "result-abc-123",
    "requestId": "req-result-001",
    "status": "success"
  },
  "data": {
    "requestId": "550e8400-e29b-41d4-a716-446655440000",
    "correlationId": "client-correlation-123",
```

```

    "status": "COMPLETED",
    "filename": "monthly_report.pdf",
    "contentType": "application/pdf",
    "fileSize": 245678,
    "processingTimeMs": 2100,
    "completedAt": "2026-01-27T09:55:03Z",
    "downloadUrl": "/api/v1/async/results/550e8400-e29b-41d4-a716-446655440000/download",
    "canAcknowledge": true,
    "documentBase64": null,
    "error": null,
    "errorCode": null
  },
  "errors": []
}

```

Response Fields:

- requestId: Unique request identifier (UUID)
- correlationId: Client-provided correlation ID
- status: Current status (QUEUED, PROCESSING, COMPLETED, FAILED)
- filename: Generated document filename (for completed requests)
- contentType: MIME type of the document
- fileSize: Document size in bytes
- processingTimeMs: Time spent generating the document
- completedAt: Completion timestamp
- downloadUrl: URL for direct binary download (for completed requests)
- canAcknowledge: Whether the result can be acknowledged/consumed
- documentBase64: Base64-encoded document content (only if includeContent=true)
- error: Error message (for failed requests)
- errorCode: Error code (for failed requests)

Download Async Result Contract

Endpoint: GET /api/v1/async/results/{requestId}/download

Description: Download the completed document as a binary file. This is a non-destructive operation - the message remains in the queue.

Authorization: Requires authentication (user can only access own requests)

Path Parameters:

- requestId (required): UUID of the async request

Response: Binary file stream with appropriate Content-Type and Content-Disposition headers

HTTP Status Codes:

- 200 OK: Document download started
- 202 Accepted: Request is still being processed
- 404 Not Found: Request not found, result not in queue, or not owned by user
- 403 Forbidden: Request belongs to another user

Response Headers:

Header	Description	Example
Content-Type	Document MIME type	application/pdf
Content-Disposition	Download filename	attachment; filename="report.pdf"
Content-Length	File size in bytes	245678

Acknowledge Async Result Contract

Endpoint: DELETE /api/v1/async/results/{requestId}

Description: Acknowledge and remove the result from the queue. This is a destructive operation - the message is permanently removed and will not be available to JMS clients.

Authorization: Requires authentication (user can only access own requests)

Path Parameters:

- requestId (required): UUID of the async request

Response: ApiResponseDto<MessageDto>

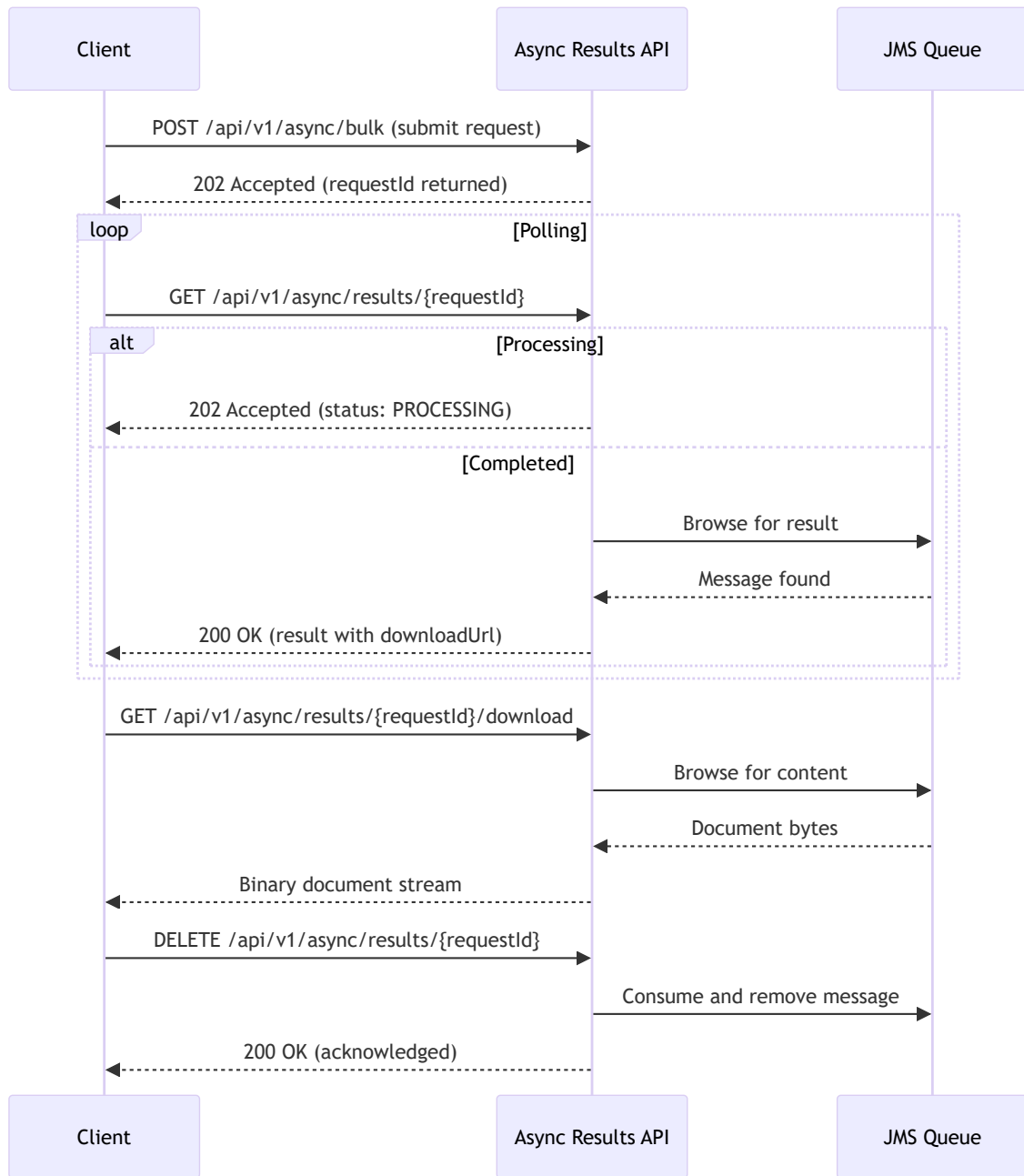
HTTP Status Codes:

- 200 OK: Result acknowledged and removed from queue
- 404 Not Found: Request not found or result not in queue
- 403 Forbidden: Request belongs to another user

```
{
  "meta": {
    "timestamp": "2026-01-27T10:00:00Z",
    "correlationId": "ack-abc-123",
    "requestId": "req-ack-001",
    "status": "success"
  },
  "data": {
    "message": "Result acknowledged and removed from queue for requestId:
    ↪ 550e8400-e29b-41d4-a716-446655440000"
  },
  "errors": []
}
```

Usage Patterns

HTTP Polling Pattern



Dual-Mode Usage

- **HTTP Polling:** Use for web applications that prefer synchronous HTTP over message queues
- **JMS Direct:** Use for high-throughput batch processing with message queue infrastructure
- **Both:** HTTP status checks with JMS consumption - browse via HTTP, consume via JMS

Important Notes:

- Acknowledgment via HTTP (DELETE) removes the message from the queue - JMS clients will no longer receive it
- If using both patterns, coordinate to avoid duplicate processing

- Unacknowledged results remain in the queue until TTL expiration or manual cleanup

Error Handling Standards

Standard Error Codes

Error Code	HTTP Status	Description	Recovery Action
AUTHENTICATION_FAILED	401	Invalid credentials	Re-authenticate with valid credentials
ACCESS_DENIED	403	Insufficient permissions	Contact administrator for access
RESOURCE_NOT_FOUND	404	Requested resource not found	Verify resource ID exists
VALIDATION_ERROR	400	Request validation failed	Fix request format and retry
INVALID_TEMPLATE_PACKAGE	422	Invalid template package upload	Verify file is a valid ZIP containing JRXML or a valid .docx file
UNSUPPORTED_FORMAT	400	Requested output format not supported by template type	DOCX templates support DOCX, PDF, HTML, and TXT output. Use a JASPER template for XLSX, RTF
INTEGRITY_VIOLATION	409	Template file integrity check failed	Regenerate digest or investigate file tampering
TEMPLATE_COMPILATION_ERROR	422	Template processing failed	Check template format and parameters
DOCUMENT_GENERATION_ERROR	500	Document generation failed	Check error message for details
TEMPLATE_FILL_ERROR	500	Failed to fill template with data	Verify data structure matches template
TEMPLATE_EXTRACTION_ERROR	500	Failed to extract template archive	Verify template package integrity
EXPORT_ERROR	500	Failed to export document	Check format compatibility
IO_ERROR	500	File I/O error during generation	Check storage permissions and space
FONT_NOT_FOUND	400	Required font is not available	Use /api/v1/templates/fonts to check available fonts
INTERNAL_ERROR	500	Server-side processing error	Retry request or contact support
RATE_LIMIT_EXCEEDED	429	Too many requests	Implement backoff and retry

Validation Error Example

```
{
  "meta": {
    "timestamp": "2025-10-10T12:31:00Z",
    "correlationId": "validation-error-456",
    "requestId": "req-validation-001",
    "status": "error"
  },
  "data": null,
  "errors": [
    {
      "code": "VALIDATION_ERROR",
      "message": "Parameter 'customerName' is required",
      "field": "parameters.customerName",
      "details": "The customerName parameter must be provided and cannot be empty"
    },
    {
      "code": "VALIDATION_ERROR",
      "message": "Invalid date format",
      "field": "parameters.invoiceDate",
      "details": "Date must be in ISO 8601 format (YYYY-MM-DD)"
    }
  ]
}
```

Document Generation Error Example

```
{
  "meta": {
    "timestamp": "2025-12-18T10:30:00Z",
    "correlationId": "generation-error-789",
    "requestId": "req-gen-001",
    "status": "error"
  },
  "data": null,
  "errors": [
    {
      "code": "TEMPLATE_FILL_ERROR",
      "message": "Failed to fill template for document 550e8400-e29b-41d4: Could not find font
        ↪ 'CustomFont'"
    }
  ]
}
```

Document generation errors include detailed messages that help identify the root cause:

- **Font issues:** “Could not find font ‘FontName’”
- **Data binding issues:** “Cannot set parameter ‘paramName’ of type X”
- **Template syntax errors:** “Error evaluating expression...”

Request/Response Headers

Required Request Headers

Header	Required	Description	Example
Authorization	Yes*	JWT Bearer token	Bearer eyJhbGciOiJIUzI1NiIs...
Content-Type	Yes**	Request content type	application/json
Accept	No	Preferred response format	application/json
X-Correlation-ID	No	Client correlation ID	client-request-abc123
X-Request-Signature	No***	HMAC-SHA256 request signature	a1b2c3d4e5f6...
X-Request-Timestamp	No***	Unix epoch timestamp (seconds)	1730992800

* Required for authenticated endpoints

** Required for requests with body

*** Required only when Request Integrity Security is enabled and enforced (see Security Considerations)

Standard Response Headers

Header	Always Present	Description	Example
Content-Type	Yes	Response content type	application/json
X-Correlation-ID	Yes	Request correlation ID	auto-gen-def456-ghi789
X-Request-ID	Yes	Unique request identifier	req-xyz789
Cache-Control	No	Caching directives	no-cache, no-store

Integration Guidelines

Best Practices

1. **Correlation ID Usage:**
 - Clients should include X-Correlation-ID header for request tracing across systems
 - Service will use client-provided correlation ID if present, or generate one automatically
 - Meaningful correlation IDs should be used to help trace requests through the entire system
2. **Error Handling:** Proper retry logic with exponential backoff should be implemented for 5xx errors
3. **Authentication:** JWT tokens should be cached and refreshed before expiration
4. **Parameter Validation:** All parameters should be validated client-side before sending requests
5. **File Size Limits:** Template uploads are limited to 50MB
6. **Timeout Configuration:** Appropriate timeouts should be set (recommended 30s for generation, 10s for queries)

Security Considerations

1. **HTTPS Only:** All communication must use HTTPS in production
2. **Token Security:** JWT tokens should be stored securely in server-side secure storage (e.g., environment variables, secret managers, secure configuration files). Tokens should not be stored in localStorage, sessionStorage, unsecured cookies, or other client-side locations accessible to JavaScript
3. **Input Sanitization:** All input data should be sanitized to prevent injection attacks
4. **Audit Compliance:** All actions are logged for audit purposes
5. **Request Integrity Security (Optional):**
 - When enabled (`security.request.signature.enabled=true`), the service can validate HMAC-SHA256 signatures on request bodies
 - Protects against Man-in-the-Middle attacks and request tampering
 - Two-phase rollout strategy:
 - **Monitoring Mode** (`enforce=false`): Logs missing/invalid signatures but allows all requests, monitor adoption rate
 - **Enforcement Mode** (`enforce=true`): Rejects requests without valid signatures (HTTP 401)
 - Required headers when enabled and enforced:
 - X-Request-Signature: `HMAC-SHA256(secretKey, timestamp + ":" + requestBody)`
 - X-Request-Timestamp: Unix epoch timestamp in seconds (must be within 300 seconds)
 - Signature validation excludes public endpoints: `/api/v1/auth/**`, `/actuator/health**`, etc.

Change Management

Versioning Strategy

- **Major Version:** Breaking changes to API contracts (e.g., v1 → v2)
- **Minor Version:** Backward-compatible feature additions
- **Patch Version:** Bug fixes and security updates

Deprecation Policy

- **Notice Period:** 90 days advance notice for breaking changes
- **Support Period:** Previous major version supported for 12 months
- **Migration Support:** Technical assistance provided during transition periods