

brightflag

MCP Documentation

For IT teams and end users.

BETA

Contents

1. Overview

2. Architecture and Data Flow

3. Tool Reference

- 3.1 resolve_filter_name
- 3.2 search_invoices
- 3.3 get_invoice
- 3.4 get_invoice_workflow
- 3.5 search_matters
- 3.6 get_entities
- 3.7 get_geos
- 3.8 get_matter_categories
- 3.9 get_users
- 3.10 get_timekeepers
- 3.11 get_invoice_review
- 3.12 get_vendors
- 3.13 get_departments
- 3.14 search_matter_attributes

4. End User Guide

5. IT and Security Reference

6. Known Limitations

7. Support and Feedback

1. Overview

Brightflag's MCP server allows you to connect your AI workspace, such as Claude, directly to Brightflag. Once connected, your AI client can access live Brightflag data and answer natural-language questions about invoices, matters, and approval workflows without you needing to switch context.

This guide covers everything end users and IT teams need to understand and operate the integration.

1.1 What is MCP?

Model Context Protocol (MCP) is an open standard for connecting AI models to external data sources. It provides a structured, auditable channel between a language model and backend APIs, replacing ad-hoc integrations with a consistent, permission-scoped model.

For Brightflag users, this means your connected AI client can call specific, read-only Brightflag endpoints on your behalf during a conversation. Each tool call is scoped, authenticated, and logged. No data is persisted between sessions.

1.2 Beta Programme

Beta notice. This MCP server is in closed beta, available to a select cohort of customers. All capabilities described here are subject to change before general availability.

The beta release exposes fourteen tools covering invoice search, invoice detail, invoice review and compliance flags, custom attribute filtering, invoice approval workflow, matter search, entity and geography lookups, matter category browsing, vendor and department lookups, user and timekeeper search, and name resolution. All operations are read-only and no data can be written or modified through MCP.

2. Architecture and Data Flow

2.1 How a Tool Call Works

When a user asks a question that requires live data, for example "Show me pending invoices from Linklaters this quarter", the following sequence occurs:

1. The user sends a message to their connected AI client, for example Claude running in claude.ai.
2. Claude identifies that a Brightflag MCP tool call is required to answer accurately.
3. Claude constructs a structured tool call, for example `resolve_filter_name` followed by `search_invoices`.
4. The MCP server authenticates the request using the user's session token and calls the Brightflag API.
5. Brightflag's API enforces tenant isolation, RBAC, and all existing access controls.
6. The structured result is returned to Claude, which synthesises a natural-language answer.
7. The user sees the answer. No data is retained by the AI client between sessions.

2.2 Authentication and Authorisation

The MCP server operates under the authenticated user's existing Brightflag session. No service accounts or elevated credentials are introduced.

- Users can only retrieve data they are already authorised to see in the Brightflag UI.
- Tenant isolation is enforced by Brightflag's API layer, not just the MCP gateway.
- Role-based access controls (RBAC) are fully respected. A reviewer sees only invoices in their queue.
- All requests are logged in Brightflag's existing audit trail.

2.3 Data and Privacy

The following summarises how data moves through the integration and what controls are in place. A full security reference is in section 5.

Control area	Detail
Data scope	The MCP server returns structured query results only. It is not possible to bulk-export Brightflag data or access records outside the scope of the specific tool call.
Data in transit	All traffic between the MCP server and Brightflag APIs is encrypted in transit over TLS 1.2+. No data is passed in URL parameters.
Data at rest	No Brightflag data is written to disk or persisted by the MCP server. Results are passed in-memory to the AI client and discarded after the session.
AI provider storage	Claude (Anthropic) does not retain conversation data between sessions under standard API terms. No Brightflag data is used for model training by default.
PII exposure	Reviewer and approver identities are limited to first and last name. Internal user IDs, email addresses, and system identifiers are stripped before results are returned.

Control area	Detail
Data residency	The MCP server is deployed in the same region as your Brightflag instance (US or EU). No data crosses regional boundaries as part of the integration.
Access control	All requests are executed under the authenticated user's existing Brightflag session. The integration cannot access data beyond that user's RBAC permissions.

3. Tool Reference

The MCP server exposes fourteen read-only tools. The table below is a quick-reference summary and full detail for each tool follows in sections 3.1 through 3.14.

Tool	Scope	Purpose
resolve_filter_name	mcp:invoice:read	Resolves a free-text name (vendor, department, entity, user) to internal IDs before searching.
search_invoices	mcp:invoice:read	Paginated, filtered invoice listing. Always the entry point for invoice queries.
get_invoice	mcp:invoice:read	Header detail for a single invoice: financials, dates, classification, matter context.
get_invoice_workflow	mcp:invoice:read	Review workflow for a single invoice: stages, reviewers, decisions, delegations.
search_matters	mcp:matter:read	Paginated, filtered matter listing.
get_entities	mcp:invoice:read	Returns trading entities (legal entities or cost centres) for the organisation.
get_geos	mcp:invoice:read	Returns available geographies for use as filters. Cached reference data.
get_matter_categories	mcp:invoice:read	Returns matter categories for the organisation, including full ID pairs.
get_users	mcp:invoice:read	Returns in-house staff with platform logins. Distinct from timekeepers.
get_timekeepers	mcp:invoice:read	Returns law-firm timekeeper rates and roles. Capped, not paginated.
get_invoice_review	mcp:invoice:read	Composite review for a single invoice: compliance flags, fee lines, and expense lines in one call.
get_vendors	mcp:invoice:read	Returns law-firm vendors visible to the authenticated user. Capped, not paginated.
get_departments	mcp:invoice:read	Returns departments visible to the authenticated user. Capped, not paginated.
search_matter_attributes	mcp:invoice:read	Resolves matter attribute display labels to fieldNames for use as filters in search_matters.

3.1 resolve_filter_name

Tool name	resolve_filter_name
Description	Resolves a free-text name into the internal IDs used by search_invoices and search_matters. Fans out in parallel across up to six reference dimensions and returns all candidates with their type and ID. Must be called before any search that

	references a named vendor, department, entity, or person.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
query	string (min 2 chars)	Yes	The name to resolve. Matched case-insensitively across all reference dimensions.
target	INVOICE MATTER	Yes	Scopes which dimensions are searched. INVOICE covers vendor, department, matter category, entity, and user. MATTER additionally includes internal_matter_lead.

Response

Field	Type	Description
query	string	Echoed from the request.
matches	array	List of matched records. Empty if no matches found.
id	string	Internal Brightflag ID for this record.
name	string	Display name of the matched record.
type	string	Dimension the match came from: vendor, department, matter_category, entity, user, or internal_matter_lead.
errors	object	Map of any reference arm failures (e.g. a downstream timeout). Other arms are unaffected.

Worked example

User asks: "Show me invoices from Linklaters"

```
// Request
{
  "query": "Linklaters",
  "target": "INVOICE"
}
```

```
// Response
{
  "query": "Linklaters",
```

```
"matches": [  
  { "id": "42", "name": "Linklaters LLP", "type": "vendor" },  
  { "id": "43", "name": "Linklaters (Frankfurt)", "type": "vendor" }  
],  
"errors": {}  
}
```

Behaviour notes

- A name matching multiple dimensions returns all candidates. The model selects by intent or asks the user to clarify.
- An empty matches list means no record was found. The tool never errors on a no-match.
- A failed reference arm, such as a downstream 503, lands in an errors map without blocking the other arms.

Trigger prompt examples

- "Find invoices from Linklaters"
- "Show me matters in the Corporate department"
- "Invoices reviewed by Jane Smith"
- "Matters led by John Doe"

3.2 search_invoices

Tool name	search_invoices
Description	Paginated, filtered invoice listing. Always the entry point for invoice queries. Returns a concise summary per invoice plus pagination metadata. Call resolve_filter_name first whenever the user references a named vendor, department, entity, or person. Never put names in the free-text query field.
OAuth scope	mcp:invoice:read

Input

Parameter	Type / Values	Notes
status	APPROVED PENDING REJECTED EBILLING_ON_HOLD BATCH_EXPORTED BATCH_PAID AWAITING_DETAILS	Omit to return all statuses.
vendorId	integer (from resolve_filter_name)	Resolve vendor name to ID first.
departmentId	integer (from resolve_filter_name)	Filter by department.
matterEntityId	integer (from resolve_filter_name)	Filter by legal entity.
reviewerUserId	integer (from resolve_filter_name)	Filter by approver.
startDate / endDate	ISO datetime, no timezone offset e.g. 2026-01-01T00:00:00	Checked against the invoice date field.
billType	STANDARD PRO_FORMA	Omit for no filter.
latestDraftOnly	boolean	When true, returns only the latest draft per invoice group.
pageSize	integer 1 to 100 (default 20)	
pageNumber	integer 1-based (default 1)	Increment while pageNumber is less than totalPages.
sortBy	invoice.uploadDate invoice.invDate invoice.invNo convertedBilledAmount	Default: invoice.uploadDate
query	string	Free-text across invoiceNumber, firmName, matterDescription only.

Response

Field	Description
invoiceId	Use this to call get_invoice or get_invoice_workflow.
invoiceNumber	Human-readable invoice reference.

Field	Description
statusName	Current lifecycle status.
firmName / vendorName	Law firm and specific vendor office.
matterId / matterDescription	Associated matter.
billedAmount / currency	Original billed amount and currency.
adjustedBilledAmount	Amount after any adjustments.
billingFlagsSummary	Summary of any compliance flags, for example: 2 flags: block billing, rate variance.
daysOutstanding	Age of the invoice in days.
totalItems / totalPages / currentPage	Pagination metadata.

Worked example

User asks: "Show me pending invoices from Linklaters uploaded this year"

```
// Request
{
  "status": "PENDING",
  "vendorId": 42,
  "startDate": "2026-01-01T00:00:00",
  "endDate": "2026-12-31T23:59:59",
  "pageSize": 20,
  "pageNumber": 1
}
```

```
// Response
{
  "totalItems": 3,
  "totalPages": 1,
  "currentPage": 1,
  "invoices": [
    {
      "invoiceId": 84201,
      "invoiceNumber": "LNK-2026-0041",
      "statusName": "PENDING",
      "firmName": "Linklaters LLP",
      "matterId": 991,
      "matterDescription": "Acquisition of MyCo Holdings",
      "billedAmount": 48500.00,
      "currency": "EUR",
      "adjustedBilledAmount": 48500.00,

```

```
    "billingFlagsSummary": "1 flag: rate variance",  
    "daysOutstanding": 12  
  }  
]  
}
```

Trigger prompt examples

- "Show me all pending invoices"
- "Invoices from Linklaters uploaded this year"
- "Rejected invoices in the Banking department"
- "Invoices waiting for approval from John Smith"
- "All paid invoices between January and March 2026"

3.3 get_invoice

Tool name	get_invoice
Description	Returns full header metadata for a single invoice. Call this after search_invoices when a user wants detailed financials, billing period, or classification for one specific invoice. Returns header facts only. Fee line items, compliance flags, and approval history are served by separate tools.
OAuth scope	mcp:invoice:read

Input

Parameter	Description
invoiceId (required, integer)	The ID from a search_invoices result.

Response

Group	Fields	Notes
Identity	invoiceId, invoiceNumber, statusName, matterId	Always present, making the response self-describing.
Financials	netFeesTotal, vatTotal, vatRate, vatOutlay, nonVatOutlay, grossTotalWithAdjustment, discount, invoiceAdjustment, flagAdjustment, disbursementTotal, disbursementAdjustment	Zero values returned; null values omitted.
Counts	lineItemCount	Total number of fee lines.
Classification	billType, processingStatus, autoProcessed, currentDraft	billType is either STANDARD or PRO_FORMA.
References	ourReference, clientReference, receiptNumber, purchaseOrderNumber, note	Omitted if not present.
Matter / firm	matterEntity, approvalRouteName	
Dates	uploadDate, completedDate, feesToDate, billingPeriodStart, billingPeriodEnd	All ISO 8601. Null fields omitted.

Routing guide

- Approval questions such as "who approved?" or "what stage?" should use get_invoice_workflow instead.
- For compliance flags, fee line items, and disbursements on this invoice, use get_invoice_review.

Worked example

User asks: "Show me the details for invoice 84201"

```
// Request
{
  "invoiceId": 84201
}
```

```
// Response
{
  "invoiceId": 84201,
  "invoiceNumber": "LNK-2026-0041",
  "statusName": "PENDING",
  "matterId": 991,
  "netFeesTotal": 43000.00,
  "vatTotal": 5500.00,
  "vatRate": 0.1277,
  "grossTotalWithAdjustment": 48500.00,
  "disbursementTotal": 0.00,
  "lineItemCount": 8,
  "billType": "STANDARD",
  "processingStatus": "IN_REVIEW",
  "matterEntity": "MyCo Holdings Ltd",
  "approvalRouteName": "Standard Legal Approval",
  "uploadDate": "2026-03-15T09:22:00",
  "billingPeriodStart": "2026-02-01",
  "billingPeriodEnd": "2026-02-28"
}
```

Trigger prompt examples

- "What's the gross total on invoice 12345?"
- "How much VAT was charged on invoice 12345?"
- "What billing period does this invoice cover?"
- "Is invoice 12345 a pro forma or a standard invoice?"
- "What adjustments have been applied to invoice 12345?"

3.4 get_invoice_workflow

Tool name	get_invoice_workflow
Description	Returns the complete review workflow for a single invoice: the ordered approval stages, each stage's consensus rule and outcome, and the reviewers in each stage with their decision, comment, amount adjustment, timestamp, and any active delegate. Also returns the calling user's available action on this invoice.
OAuth scope	mcp:invoice:read

Input

Parameter	Description
invoiceId (required, integer)	The ID from a search_invoices result.

Response

Field	Description
invoiceId	Echoed from the request, making the result self-describing.
reviewAction	The calling user's available action. VIEW means read-only. REVIEW means the user can approve or reject at the current stage. FINAL_REVIEW means the user can give the final decision. REVERT means the user can reopen a completed decision.
workflowStages[]	Ordered list of review stages.
stageNumber	Stage sequence (1, 2, and so on).
consensusType	UNANIMOUS means all reviewers must approve. SIMPLE means one approval advances the stage.
consensusResult	The overall outcome for this stage: PENDING, APPROVED, or REJECTED.
stageReviews[]	The reviewers assigned to this stage.
reviewer	First and last name only. Internal user ID is never exposed.
status	PENDING, APPROVED, or REJECTED.
comment	The reviewer's comment if provided. Omitted if absent.
adjustment	Amount adjustment applied by this reviewer. Omitted if zero or absent.
reviewedAt	ISO 8601 UTC timestamp of the review action. Omitted if the reviewer has not yet acted.
delegate	First and last name of the active delegate, if delegation is in effect.

Behaviour notes

- If an invoice has no review stages yet, an empty stage list is returned rather than an error.

- Reviewer and delegate identities are returned as first and last name only. Internal user IDs are stripped at the gateway.
- Empty reviewer fields such as comment, adjustment, timestamp, and delegate are omitted from the response.

Worked example

User asks: "Who still needs to approve invoice 84201?"

```
// Request
{
  "invoiceId": 84201
}
```

```
// Response
{
  "invoiceId": 84201,
  "reviewAction": "VIEW",
  "workflowStages": [
    {
      "stageNumber": 1,
      "consensusType": "SIMPLE",
      "consensusResult": "APPROVED",
      "stageReviews": [
        {
          "reviewer": "Jane Smith",
          "status": "APPROVED",
          "reviewedAt": "2026-03-16T11:05:00Z"
        }
      ]
    },
    {
      "stageNumber": 2,
      "consensusType": "UNANIMOUS",
      "consensusResult": "PENDING",
      "stageReviews": [
        { "reviewer": "Tom Clarke", "status": "APPROVED", "reviewedAt": "2026-03-17T09:10:00Z" },
        { "reviewer": "Sarah Lee", "status": "PENDING" }
      ]
    }
  ]
}
```

Trigger prompt examples

- "Who still needs to approve invoice 84213?"

- "What stage is invoice 12345 at in the review process?"
- "Can I approve invoice 84213, or has someone delegated it?"
- "Which stage requires unanimous approval on this invoice?"

3.5 search_matters

Tool name	search_matters
Description	Paginated, filtered matter listing mirroring the Matter Search page. Always the entry point for matter queries. Call resolve_filter_name first whenever the user references a named firm, department, entity, or person. Returns a concise matter summary per result plus pagination metadata.
OAuth scope	mcp:matter:read

Input

Parameter	Type / Values	Notes
matterStatus	OPEN CLOSED REQUESTED REJECTED_REQUEST	Omit to return all. OPEN and CLOSED are most common.
matterCategoryId	integer (from resolve_filter_name)	Filter by matter category.
matterEntityId	integer (from resolve_filter_name)	Filter by trading entity.
departmentId	integer (from resolve_filter_name)	Filter by department.
firmId	integer (from resolve_filter_name)	Filter by law firm.
vendorId	integer (from resolve_filter_name)	Filter by vendor.
internalMatterLeadUserId	integer (from resolve_filter_name)	Filter by matter lead.
matterParticipantId	integer	Filter to matters where this user is a participant.
geold	integer (from get_geos)	Filter by geographic region.
matterIds	integer[]	Look up a specific set of matters by ID.
startDate / endDate	Date string, e.g. 2026-01-01	Matter date range filter.
freeText	string	Free-text across matter reference, name and related fields. Never put names here.
attributeFieldName	string (from search_matter_attributes)	No
attributeValue	string	No
pageSize	integer 1 to 100 (default 20)	
pageNumber	integer 1-based (default 1)	Increment while pageNumber is less than totalPages.
sortBy	uploadDate matterRef matterName customerMatterID	Default: uploadDate.

Response

Field	Description
matterId	Unique identifier for the matter.
matterReference	Human-readable matter reference.
matterName	Matter description and name.
status	OPEN, CLOSED, REQUESTED, or REJECTED_REQUEST.
entityName	The trading entity this matter belongs to.
departmentName	The department responsible for this matter.
internalMatterLeadUserName	Username of the internal matter lead.
attributes	array
totalItems / totalPages / currentPage	Pagination metadata.

Worked example

User asks: "Show me all open matters for Linklaters in the Banking category"

```
// Request
{
  "matterStatus": "OPEN",
  "firmId": 42,
  "matterCategoryId": 7,
  "pageSize": 20,
  "pageNumber": 1,
  "sortBy": "matterName"
}
```

```
// Response
{
  "totalItems": 2,
  "totalPages": 1,
  "currentPage": 1,
  "matters": [
    {
      "matterId": 991,
      "matterReference": "BF-2024-0112",
      "matterName": "Acquisition of MyCo Holdings",
      "status": "OPEN",
      "entityName": "MyCo Holdings Ltd",
      "departmentName": "Corporate",
      "internalMatterLeadUserName": "j.doe",
      "attributes": [
```

```
{
  "type": "Drop Down",
  "Text": {
    { "fieldName": "matterRisk", "name": "Matter Risk", "value": "High",
      { "fieldName": "region", "name": "Region", "value": "EMEA", "type":
    }
  },
  {
    "matterId": 1044,
    "matterReference": "BF-2025-0019",
    "matterName": "Refinancing Facility Agreement",
    "status": "OPEN",
    "entityName": "MyCo Holdings Ltd",
    "departmentName": "Banking",
    "internalMatterLeadUserName": "s.chen",
    "attributes": []
  }
}
```

Trigger prompt examples

- "Show me all open matters for Linklaters"
- "Matters in the Banking category"
- "Closed matters in the United Kingdom"
- "Matters where John Smith is the internal lead"

- "Open matters for entity MyCo in the Corporate department"

3.6 get_entities

Tool name	get_entities
Description	Returns the list of trading entities (legal entities or cost centres) for the authenticated user's organisation. Used when the user wants to browse available entities, or when a direct lookup is needed that goes beyond what resolve_filter_name returns. Results are capped at 150; supply a nameFilter to narrow a large list.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
nameFilter	string	No	Optional case-insensitive entity name filter. Omit to list all entities up to the cap. If the result is truncated, supply or tighten this filter.

Response

Field	Type	Description
truncated	boolean	True when the result has been capped and more entities exist. Narrow with nameFilter rather than paging.
values	array	List of matching entities.
id	string	Entity ID. Pass as matterEntityId in search_invoices or search_matters.
name	string	Display name of the entity.

Worked example

User asks: "What legal entities do we have?"

```
// Request
{
  "nameFilter": "MyCo"
}
```

```
// Response
{
  "truncated": false,
  "values": [
    { "id": "3", "name": "MyCo UK Ltd" },
  ]
}
```

```
{ "id": "7", "name": "MyCo Ireland Ltd" },  
  { "id": "12", "name": "MyCo US Inc." }  
]  
}
```

Behaviour notes

- Results are capped at 150 rows, not paginated. When truncated is true, narrow the query using nameFilter.
- Disabled entities are excluded from results.
- Use the id field from results as matterEntityId in search_invoices or search_matters.

Trigger prompt examples

- "What legal entities do we have?"
- "Is there an entity named MyCo?"
- "Show me all entities matching UK"

3.7 get_geos

Tool name	get_geos
Description	Returns the list of geographies (countries and regions) available in the system. Used when the user wants to browse available geographies or resolve a geo name to its ID for use as a filter in other tools. This is shared, tenant-agnostic reference data cached in-memory by the gateway and refreshed at most once per day.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
nameFilter	string	No	Optional case-insensitive geography name filter applied over the cached list. Omit to list all geographies. The full list is bounded and returned in one call.

Response

Field	Type	Description
truncated	boolean	Always false for this tool. The full geography list is always returned in one call.
values	array	List of matching geographies, sorted alphabetically by name.
id	string	Geography ID. Pass as geold in search_matters.
name	string	Display name of the geography.

Worked example

User asks: "Is there a UK geography I can filter by?"

```
// Request
{
  "nameFilter": "United Kingdom"
}
```

```
// Response
{
  "truncated": false,
  "values": [
    { "id": "77", "name": "United Kingdom" }
  ]
}
```

```
}
```

Behaviour notes

- Results are never paginated or capped. The full geography list is always returned in a single call.
- Data is cached in-gateway and refreshed at most once per day. If a refresh fails, the last known-good list is returned.
- Use the id field from results as geold in search_matters.

Trigger prompt examples

- "What geographies can I choose from?"
- "List the countries available as filters"
- "Is there a UK geography?"

3.8 get_matter_categories

Tool name	get_matter_categories
Description	Returns matter categories for the organisation. Used when the user wants to browse available categories, or when a search requires the full category ID pair (id and categoryCustomID). Note that resolve_filter_name returns only the id field; this tool returns the complete record. Results are capped at 150; supply a nameFilter to narrow a large list.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
nameFilter	string	No	Optional case-insensitive matter category name filter. Omit to list all categories up to the cap. If the result is truncated, supply or tighten this filter.

Response

Field	Type	Description
truncated	boolean	True when the result has been capped and more categories exist. Narrow with nameFilter rather than paging.
values	array	List of matching matter categories.
id	string	Matter category ID. Pass as matterCategoryId in search_invoices or search_matters.
name	string	Display name of the matter category.

Worked example

User asks: "What matter categories do we have?"

```
// Request
{
  "nameFilter": "lit"
}
```

```
// Response
{
  "truncated": false,
  "values": [
```

```
{ "id": "201", "name": "Litigation" },  
  { "id": "208", "name": "Commercial Litigation" }  
]
```

Behaviour notes

- Results are capped at 150 rows, not paginated. When truncated is true, narrow the query using nameFilter.
- Use the id field from results as matterCategoryId in search_invoices or search_matters.
- Use this tool when resolve_filter_name alone is not sufficient and the full category record is required.

Trigger prompt examples

- "What matter categories do we have?"
- "Show me all litigation-related categories"
- "Is Banking a valid matter category?"

3.9 get_users

Tool name	get_users
Description	Returns in-house staff with platform logins. Distinct from timekeepers, which are firm-side billers. Used when the user wants to browse or look up internal users, or when resolve_filter_name is not sufficient and a direct lookup by name or ID is needed. Results are capped at 150; supply a nameFilter to narrow a large list.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
nameFilter	string	No	Optional case-insensitive filter matched server-side against user name and email. Omit to list all active users up to the cap. If the result is truncated, supply or tighten this filter.

Response

Field	Type	Description
truncated	boolean	True when the result has been capped and more users exist. Narrow with nameFilter rather than paging.
values	array	List of matching users.
id	string	User ID. Pass as reviewerUserId in search_invoices or internalMatterLeadUserId in search_matters.
name	string	Display name. Falls back to username when no full name is set.
username	string	The user's login email address.

Worked example

User asks: "Find users called Jordan"

```
// Request
{
  "nameFilter": "jordan"
}
```

```
// Response
{
  "truncated": false,
```

```
"values": [  
  { "id": "9021", "name": "Jordan Mensah", "username":  
    "jordan.mensah@myco.com" },  
  { "id": "9044", "name": "jokafor@myco.com", "username":  
    "jokafor@myco.com" }  
]
```

Behaviour notes

- Results are capped at 150 rows, not paginated. When truncated is true, narrow the query using nameFilter.
- Only active users are returned. Inactive or disabled accounts are excluded.
- A user's display name falls back to their username when no full name has been set on their account.
- Use the id field as reviewerUserId in search_invoices or internalMatterLeadUserId in search_matters.

Trigger prompt examples

- "List the users I can filter by"
- "Look up user John Smith"
- "Find users called Jordan"

3.10 get_timekeepers

Tool name	get_timekeepers
Description	Searches the customer's law-firm timekeepers and returns their current approved or pending rates including rate amount, currency, role, rate category (MATTER or GENERAL), and the billing law-firm vendor ID. Used when the user asks about a law-firm timekeeper's rate, role, or identity. Results are capped, not paginated. A row is one timekeeper-role-rate combination, so a timekeeper with multiple roles appears more than once.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
query	string	No	Optional free-text term matched case-insensitively against a timekeeper's first name, surname, LEDES timekeeper ID, or full name. Omit to list all timekeepers up to the cap. If the result is truncated, supply or tighten this term.

Response

Field	Type	Description
truncated	boolean	True when the result has been capped and more rows exist. Narrow with query rather than paging.
rates	array	List of timekeeper-role-rate rows. One person may appear multiple times.
timekeeperId	integer	Internal timekeeper identifier.
firstName	string	Timekeeper first name.
lastName	string	Timekeeper last name.
role	string	Timekeeper role, e.g. Partner, Associate.
rateAmount	number	Rate amount. Omitted when no rate is set for this row.
currency	string	Currency of the rate amount. Omitted when no rate is set.
rateStatus	string	APPROVED or PENDING only. Rejected rates are never returned.
rateCategory	string	MATTER (applies to a specific matter) or GENERAL (applies across all matters).
vendorId	integer	ID of the billing law-firm vendor this timekeeper belongs to.

Worked example

User asks: "What rate does timekeeper Jane Berckenkamp charge?"

```
// Request
{
  "query": "Berckenkamp"
}
```

```
// Response
{
  "truncated": false,
  "rates": [
    {
      "timekeeperId": 42,
      "firstName": "Jane",
      "lastName": "Berckenkamp",
      "role": "Partner",
      "rateAmount": 500.00,
      "currency": "USD",
      "rateStatus": "APPROVED",
      "rateCategory": "GENERAL",
      "vendorId": 1
    },
    {
      "timekeeperId": 42,
      "firstName": "Jane",
      "lastName": "Berckenkamp",
      "role": "Partner",
      "rateAmount": 520.00,
      "currency": "USD",
      "rateStatus": "PENDING",
      "rateCategory": "MATTER",
      "vendorId": 1
    }
  ]
}
```

Behaviour notes

- A row represents one timekeeper-role-rate combination, not one person. A timekeeper with multiple roles or both an approved and pending rate appears as multiple rows.
- Results are capped at a fixed maximum, not paginated. When truncated is true, narrow using the query parameter.
- Only APPROVED and PENDING rates are returned. Rejected rates are excluded.
- rateAmount and currency are omitted from a row when no rate has been set for that role.
- Distinct from get_users, which returns in-house client staff rather than law-firm timekeepers.

Trigger prompt examples

- "What rate does timekeeper Jane Berckenkamp charge?"

- "Who is timekeeper ARE123?"
- "Show me the firm's timekeepers and their rates"
- "Find timekeeper Smith"

3.11 get_invoice_review

Tool name	get_invoice_review
Description	Composite review for a single invoice. In one call returns all three review sections: the AI-generated compliance flags raised on the invoice, its fee (time) line items, and its expense/disbursement line items. Each section is access-controlled independently so a partial result is returned when the user can only see some sections. Line volume is capped rather than paginated. Use flagId to narrow fee and expense lines to those a specific flag was raised against.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
invoiceId	integer	Yes	The ID of the invoice to review. Obtain from a search_invoices or get_invoice result.
flagId	integer	No	Optional. Narrow the returned fee and expense lines to those a specific flag was raised against. Use when lines come back truncated to focus on one flag. Does not affect the flags section.

Response

Field	Type	Description
invoiceId	integer	Echoed from the request.
flags	object	Compliance flags section.
flagsList	array	List of compliance flags raised on this invoice.
flagId	integer	Unique flag identifier. Pass as flagId to narrow lines.
flagName	string	Display name of the flag rule.
flagDescription	string	Description of why this flag was raised.
flagCategory	string	Category of the flag, e.g. TIMEKEEPERS, BILLING.
amount	number	Amount billed that the flag applies to.
suggestedSaving	number	Suggested reduction amount. Omitted when zero.
detaillineId	integer	ID of the fee or expense line the flag was raised against.
fees	object	Fee (time) line items section.
truncated	boolean	True when the line cap was reached. Narrow using flagId.
returnedLines	integer	Number of lines returned when truncated is true.
lines	array	List of fee line items.

Field	Type	Description
detaillineId	integer	Line identifier. Matches detaillineId on flags.
description	string	Narrative description of the work performed.
date	string	Date of the fee entry (ISO 8601).
timekeeper	string	Name of the billing timekeeper.
role	string	Timekeeper role, e.g. Partner, Associate.
durationMinutes	integer	Time billed in minutes.
rate	number	Hourly rate applied.
amount	number	Gross amount billed.
adjust	number	Adjustment amount. Omitted when zero.
newLineTotal	number	Net amount after adjustment.
flagIds	array	IDs of any flags raised against this line.
expenses	object	Expense/disbursement line items section.
truncated	boolean	True when the line cap was reached.
lines	array	List of expense line items.
disbursementInvoiceId	integer	Expense line identifier.
description	string	Description of the expense.
date	string	Date of the expense (ISO 8601).
amount	number	Amount charged.
categoryName	string	Expense category, e.g. Courier, Filing Fee.
flagIds	array	IDs of any flags raised against this expense.
errors	object	Map of section names to error messages for sections the user cannot access. Other sections are unaffected.

Worked example

User asks: "Are there any compliance issues on invoice 84201?"

```
// Request
{
  "invoiceId": 84201
}
```

```
// Response
{
  "invoiceId": 84201,
  "flags": {
    "flagsList": [
      {
        "flagId": 88012,
        "flagName": "Internal Communications",
        "flagDescription": "Multiple timekeepers billed for the same internal
meeting.",
        "flagCategory": "TIMEKEEPERS",
        "amount": 1200.00,
        "suggestedSaving": 300.00,
        "detaillineId": 990001
      }
    ]
  },
  "fees": {
    "truncated": false,
    "lines": [
      {
        "detaillineId": 990001,
        "description": "Review and revise draft motion; correspondence with co-
counsel.",
        "date": "2026-03-14",
        "timekeeper": "Jane Roe",
        "role": "Partner",
        "durationMinutes": 150,
        "rate": 480.00,
        "amount": 1200.00,
        "adjust": -300.00,
        "newLineTotal": 900.00,
        "flagIds": [88012]
      }
    ]
  },
  "expenses": {
    "truncated": false,
    "lines": [
      {
        "disbursementInvoiceId": 550021,
        "description": "Courier - overnight delivery to co-counsel",
        "date": "2026-03-15",
        "amount": 48.00,
        "categoryName": "Courier",
        "flagIds": []
      }
    ]
  }
}
```

Behaviour notes

- Each of the three sections (flags, fees, expenses) is access-controlled independently. If the user cannot access a section, it is omitted and listed in the errors map; the other sections are returned normally.
- Fee and expense lines are capped at a fixed maximum per section, not paginated. When truncated is true, use flagId to narrow lines to those a specific flag was raised against.
- When flagId is supplied, only fee and expense lines that flag was raised against are returned. The flags section is always returned in full regardless of flagId.

Trigger prompt examples

- "Are there any billing compliance issues on invoice 12345?"
- "Show me the fee breakdown for this invoice"
- "What timekeeper rates were charged on this invoice?"
- "Show me all disbursements and expenses on this invoice"
- "Why was invoice 12345 flagged?"
- "Show me the line items the Internal Communications flag hit on invoice 84201"

3.12 get_vendors

Tool name	get_vendors
Description	Returns the list of law-firm vendors visible to the authenticated user. Used when the user wants to browse available vendors, or when resolve_filter_name alone is not sufficient and a direct vendor lookup is needed. Results are capped at 150; supply a nameFilter to narrow a large list.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
nameFilter	string	No	Optional case-insensitive vendor name filter applied server-side. Omit to list all vendors up to the cap. If the result is truncated, supply or tighten this filter.

Response

Field	Type	Description
truncated	boolean	True when the result has been capped and more vendors exist. Narrow with nameFilter rather than paging.
returnedItems	integer	Number of vendors returned when truncated is true.
values	array	List of matching vendors.
id	string	Vendor ID. Pass as firmId in search_invoices or search_matters.
name	string	Display name of the vendor. Uses custom vendor name when set, otherwise the firm name.

Worked example

User asks: "Is Grogan and Dalton a vendor I can filter by?"

```
// Request
{
  "nameFilter": "Grogan"
}
```

```
// Response
{
  "truncated": false,
  "values": [
```

```
{ "id": "1043", "name": "Grogan & Dalton LLP" }  
]  
}
```

Behaviour notes

- Results are capped at 150 rows, not paginated. When truncated is true, narrow the query using nameFilter.
- A vendor's display name uses the custom vendor name when set, and falls back to the firm name when no custom name exists.
- Use the id field from results as firmId in search_invoices or search_matters.

Trigger prompt examples

- "Show me all vendors"
- "Which vendors start with Gro?"
- "Is Grogan and Dalton a vendor I can filter by?"

3.13 get_departments

Tool name	get_departments
Description	Returns the list of departments visible to the authenticated user. Department-limited users are automatically scoped to their own departments. Used when the user wants to browse available departments, or when a direct lookup is needed beyond what resolve_filter_name returns. Results are capped at 150; supply a nameFilter to narrow a large list.
OAuth scope	mcp:invoice:read

Input

Parameter	Type	Required	Description
nameFilter	string	No	Optional case-insensitive department name filter applied client-side. Omit to list all departments up to the cap. If the result is truncated, supply or tighten this filter.

Response

Field	Type	Description
truncated	boolean	True when the result has been capped and more departments exist. Narrow with nameFilter rather than paging.
values	array	List of matching departments.
id	string	Department ID. Pass as departmentId in search_invoices or

Field	Type	Description
		search_matters.
name	string	Display name of the department.

Worked example

User asks: "Is there a Corporate department?"

```
// Request
{
  "nameFilter": "Corporate"
}
```

```
// Response
{
  "truncated": false,
  "values": [
    { "id": "12", "name": "Corporate" }
  ]
}
```

Behaviour notes

- Results are capped at 150 rows, not paginated. When truncated is true, narrow the query using nameFilter.
- Department-limited users automatically receive only their own departments, regardless of the nameFilter supplied.
- Use the id field from results as departmentId in search_invoices or search_matters.

Trigger prompt examples

- "What departments do we have?"
- "Is there a Corporate department?"
- "Show me all departments matching compliance"

3.14 search_matter_attributes

Tool name	search_matter_attributes
Description	Returns the customer's full matter custom attribute catalog so the connected AI can resolve a display label (e.g. 'Matter Risk') to the stable fieldName (e.g. matterRisk) that search_matters' attributeFieldName filter requires.
OAuth scope	mcp:invoice:read

Input

Parameter	Description
nameFilter (optional, string)	Case-insensitive filter matched against attribute display names and fieldNames. Omit to return all attributes up to the cap. When truncated is true, supply or tighten this filter.

Response

Field	Description
truncated	True when the result has been capped at 150 and more attributes exist. Narrow with nameFilter.
returnedItems	Number of attributes returned when truncated is true.
attributes	List of matching attribute definitions.
fieldName	Stable system key. Pass as attributeFieldName in search_matters.
name	Human-readable display label as shown in the Brightflag UI.
type	Attribute type, e.g. Drop Down, Text, Date.
matterTemplate	Present only for template-scoped attributes. Omitted for core attributes that apply to every matter.

Worked example

User asks: "What is the fieldName for the Matter Risk attribute?"

```
// Request { "nameFilter": "Matter Risk" }
// Response { "truncated": false, "attributes": [ { "fieldName": "matterRisk", "name": "Matter Risk", "type": "Drop Down" } ] }
```

Behaviour notes

- Results are capped at 150 rows, not paginated. When truncated is true, narrow using nameFilter.
- nameFilter matches case-insensitively against both the display name and the fieldName.
- Core attributes omit the matterTemplate field. Template-scoped attributes include the owning template's name.
- A nameFilter matching nothing returns an empty attributes list, not the unfiltered catalog.
- Use this tool to discover attribute fieldNames before filtering search_matters by attributeFieldName or attributeValue.

Trigger prompt examples

- "What custom attributes can I filter matters by?"
- "Is there a Region attribute I can filter matters by?"
- "What is the fieldName for the Matter Risk attribute?"
- "Show me all available matter attributes"

4. End User Guide

The MCP integration is enabled at the account level by your Brightflag administrator. Once connected to your AI workspace, you can access live Brightflag data by asking questions in plain English. No special commands or syntax are required.

4.1 What You Can Ask

Invoice search

- "Show me all pending invoices from this month"
- "Which invoices from Linklaters are still waiting for approval?"
- "Find rejected invoices in the Banking department in Q1 2026"
- "How many invoices do we have on hold?"

Invoice detail

- "What's the net fees total on invoice INV-2026-0042?"
- "Break down the VAT and disbursements on invoice 12345"
- "What billing period does this invoice cover?"
- "Is invoice 12345 a pro forma or standard invoice?"

Approval workflows

- "Who still needs to approve invoice 84213?"
- "What stage is invoice 12345 at in the review process?"
- "Has John Smith delegated his review on any open invoices?"
- "Which stage requires unanimous approval on this invoice?"

Matter search

- "Show me all open matters for MyCo Holdings"
- "Which matters in the Corporate department are still open?"
- "Find matters led by Jane Doe"
- "List closed matters from 2025 in the Banking category"

Entities and geographies

- "What legal entities do we have?"
- "Is there an entity named MyCo UK?"
- "What geographies can I filter by?"
- "Is there a United Kingdom geography?"

Matter categories

- "What matter categories do we have?"
- "Show me all litigation-related categories"
- "Is Banking a valid matter category?"

Users

- "List the users I can filter invoices by"
- "Find users called Jordan"
- "Look up user John Smith"

Timekeepers

- "What rate does timekeeper Jane Berckenkamp charge?"
- "Who is timekeeper ARE123?"
- "Show me all timekeepers and their rates"
- "Find timekeeper Smith"

Invoice compliance and line items

- "Are there any billing compliance issues on invoice 12345?"
- "Show me the fee breakdown for this invoice"
- "Why was invoice 12345 flagged?"
- "Show me all disbursements and expenses on this invoice"

Vendors and departments

- "Show me all vendors"
- "Is Grogan and Dalton a vendor I can filter by?"
- "What departments do we have?"
- "Is there a Corporate department?"

4.2 How Name Resolution Works

When you mention a vendor, firm, department, entity, or person by name, the connected AI will automatically attempt to resolve the name to the correct internal Brightflag ID before searching. For common entities this happens via `resolve_filter_name`. For reference data such as geographies, matter categories, entities, users, timekeepers, vendors, and departments, the dedicated lookup tools can be used directly when browsing or when a more specific lookup is needed.

Example:

- You say: "Show me Linklaters invoices from last quarter"
- Internally, the AI calls `resolve_filter_name` with "Linklaters", finds the vendor ID, then calls `search_invoices` with that ID
- You see a summary of matching invoices, grounded in live data

Note. If a name matches more than one record, for example a person who is also listed as a vendor, the connected AI will ask you to clarify before proceeding.

4.3 Limitations to Be Aware Of

Limitation	Detail
Read-only	MCP cannot approve, reject, create, or modify any data in Brightflag. All operations are read-only.
Your permissions apply	The connected AI can only retrieve data you are already authorised to see in Brightflag. RBAC and tenant boundaries are fully enforced.
No historical audit	The MCP tools return current state only, not a full audit log of all historical changes.
Search result pagination	Invoice and matter searches return paginated results. If a query matches a very large number of records, narrow your search using filters.
Reference data caps	Lookup tools (get_entities, get_users, get_matter_categories, get_timekeepers, get_vendors, get_departments, search_matter_attributes) return up to 150 results per call rather than paginating. Use the nameFilter or query parameter to narrow large lists.
Beta status	Some edge cases may not yet be handled gracefully. Report unexpected behaviour via your Brightflag customer success contact.

5. IT and Security Reference

This section is written for IT and security teams evaluating the Brightflag MCP integration for internal approval. It covers authentication model, authorisation boundaries, data classification, network controls, audit capability, and third-party data handling.

5.1 Integration Architecture Summary

The Brightflag MCP server acts as a controlled API gateway between an authorised AI client and the Brightflag platform. It does not introduce a new data store, a new identity system, or any persistent connection. Each tool call is a discrete, authenticated, read-only HTTP request made on behalf of a specific user.

Property	Value
Protocol	HTTPS / REST over TLS 1.2+
HTTP methods	GET only. No POST, PUT, PATCH, or DELETE operations are exposed.
Authentication	OAuth 2.0 bearer token (existing Brightflag user session). No service accounts or shared credentials.
Direction of traffic	Outbound from MCP server to Brightflag APIs only. No inbound connections to customer infrastructure required.
Persistence	Stateless. No Brightflag data is written to disk or cached by the MCP server.
Deployment region	Co-located with the customer's Brightflag instance (US or EU). No cross-regional data transfer.

5.2 Authentication and Session Model

The MCP server does not maintain its own credential store. All requests are authenticated using the end user's active Brightflag OAuth 2.0 session token, which is passed as a Bearer token in the Authorization header of each API call.

- No service accounts, shared API keys, or long-lived tokens are used.
- Tokens are scoped to the individual user's session and expire in line with Brightflag's standard session timeout policy.
- Token values are never logged, written to disk, or included in URL query strings.
- If a session expires mid-interaction, the API returns a 401 and no data is returned. Re-authentication is required through the standard Brightflag login flow.

5.3 Authorisation and Access Control

The MCP integration does not introduce new permission levels or bypass existing controls. All data access is governed entirely by Brightflag's existing role-based access control (RBAC) model.

Control	Detail
RBAC enforcement	The Brightflag API enforces the authenticated user's role and permissions on every request. The MCP server cannot retrieve data the user cannot already access in the Brightflag UI.
OAuth scopes	Two read-only scopes are required: mcp:invoice:read and mcp:matter:read. These are additive to, not replacements for, existing Brightflag permission checks.
Tenant isolation	Brightflag's API layer enforces tenant-scoped queries via application-level filters and JPA/Hibernate interceptors. Cross-tenant data access is not possible regardless of tool called.
Privilege escalation	Not possible. The integration operates entirely within the permissions of the authenticated user. There are no admin-level or elevated-access endpoints exposed via MCP.
Write access	None. All fourteen MCP tools are read-only. No create, update, approve, reject, or delete operations are exposed.

5.4 Data Classification and Handling

The MCP server returns structured query results scoped to the requesting user's permissions. The following controls apply to all data in transit through the integration.

Classification area	Detail
Data scope	Tool responses contain structured summaries of invoice and matter records. Bulk export of raw database content is not possible through any MCP tool.
PII minimisation	Reviewer and approver identities are restricted to first and last name only. Internal user IDs, email addresses, usernames, and system identifiers are stripped at the MCP gateway before results are returned to the AI client.
Sensitive fields	Financial totals, VAT amounts, and billing data are included in invoice responses as these are the core use case. Access is gated by the user's existing invoice RBAC permissions.
Data retention	The MCP server holds no state. Results are passed in-memory to the AI client and not persisted. No Brightflag data is written to disk by the MCP layer.
AI client retention	Under standard Anthropic API terms, Claude does not persist conversation data between sessions and does not use API data for model training. Customers requiring specific data processing terms should review their Anthropic DPA or contact Brightflag.

5.5 Network and Transport Security

Control	Detail
Encryption in transit	All traffic between the MCP server and Brightflag APIs is encrypted over TLS 1.2 or higher. TLS 1.0 and 1.1 are not supported.

Control	Detail
Encryption at rest	No Brightflag data is written to disk by the MCP server. This control is not applicable to the MCP layer; Brightflag's platform-level encryption at rest applies to the underlying data store.
Inbound firewall changes	None required. The MCP server initiates outbound connections to Brightflag APIs only. No new inbound ports or firewall rules are needed on the customer's network perimeter.
Certificate validation	Standard TLS certificate validation is enforced. Self-signed or expired certificates are rejected.
Token transport	OAuth tokens are passed in the Authorization HTTP header exclusively. They are never included in URL query strings, request bodies, or log output.

5.6 Audit and Logging

All MCP-initiated requests to the Brightflag API are captured in Brightflag's existing audit log infrastructure. No additional logging tooling is required.

Log field	Detail
User identity	The authenticated Brightflag user who initiated the request.
Timestamp	UTC timestamp of the request.
Resource accessed	The Brightflag API resource called (e.g. invoice search, matter lookup).
HTTP status code	The outcome of the request (200, 401, 403, 500, etc.).

- Logs are available to Brightflag tenant administrators through the standard audit log interface.
- Log retention follows Brightflag's platform-level retention policy. Contact your Brightflag account team for specific retention periods.
- MCP requests are not distinguishable from standard Brightflag UI requests in the audit log. They appear as API calls made by the authenticated user.

5.7 Third-Party Data Processor: Anthropic

When using Claude as the connected AI client, Anthropic acts as a third-party data processor. The following applies to the Anthropic API integration.

Topic	Detail
Legal basis	Data processing is governed by the Anthropic Data Processing Agreement (DPA) accepted at onboarding. Customers with specific contractual requirements should review their Anthropic agreement or contact Brightflag.
Data use for training	Anthropic does not use API conversation data to train models by default under standard API terms. This is distinct from Anthropic's consumer product (Claude.ai), which has different terms.

Topic	Detail
Session data	Claude does not persist Brightflag data between separate user sessions under standard API operation.
Sub-processors	Anthropic's current sub-processor list is published at anthropic.com/legal . Customers subject to sub-processor notification requirements should subscribe to Anthropic's updates.
Data transfer	If your Brightflag instance is EU-hosted, data passed to the Anthropic API may be processed outside the EU. Review the Anthropic DPA for applicable transfer mechanisms (e.g. Standard Contractual Clauses).

6. Known Limitations

6.1 Current Beta Limitations

Limitation	Detail
Read-only	No write operations. Invoices and matters cannot be approved, rejected, or updated via the MCP tools.
Pagination on large sets	The connected AI will not auto-paginate through very large result sets, and reference lookups cap at 150 rows rather than paginating. Use specific filters or narrowing terms to keep results manageable.
Beta stability	As a pre-GA release, edge cases and error messages may be less polished. Please report issues to your customer success contact.

7. Support and Feedback

This integration is in closed beta and your feedback is essential to shaping the GA release.

Type	Contact
End user questions and unexpected behaviour	Raise with your internal Brightflag administrator or contact Brightflag Customer Success.
IT and security queries	Contact your Brightflag account team or email security@brightflag.com .
Feature requests and feedback	Use the in-product feedback mechanism or contact your Customer Success Manager.
Critical bugs	Contact Brightflag support with the invoice or matter ID and the query that triggered the issue.