

# CCRS — User Manual

Comprehensive guide to the Contract & Compliance Review System (CCRS) — Digittal Group's platform for managing the full contract lifecycle from drafting through execution and archival.

## Audience

- **Internal staff** — System Admins, Legal, Commercial, Finance, Operations, Audit
- **Board/executive stakeholders**
- **External counterparties and vendors**

## Table of Contents

1. [Platform Overview](#)
2. [Getting Started](#)
3. [Contract Lifecycle](#)
4. [Counterparty Management](#)
5. [Signing & E-Signatures](#)
6. [Workflow Templates](#)
7. [AI Analysis & Redlining](#)
8. [Reports & Analytics](#)
9. [Bulk Operations](#)
10. [Notifications & Reminders](#)
11. [Organization Setup](#)
12. [Vendor Portal](#)
13. [External Signing Guide](#)
14. [Role Reference Matrix](#)
15. [Compliance & Audit](#)

## Quick Links

- [Role Reference Matrix](#) — permissions and access levels for every role
  - [External Signing Guide](#) — step-by-step instructions for external signatories
  - [Vendor Portal](#) — portal access for counterparties and vendors
- 

## Platform Overview

### What is CCRS?

CCRS (Contract & Compliance Review System) is Digittal Group's centralised platform for managing the full lifecycle of contracts and merchant agreements. Built on Laravel 12 and Filament 3, CCRS brings together contract drafting, electronic signing, AI-powered analysis, compliance tracking, and vendor self-service into a single web application backed by Azure AD authentication and role-based access control.

---

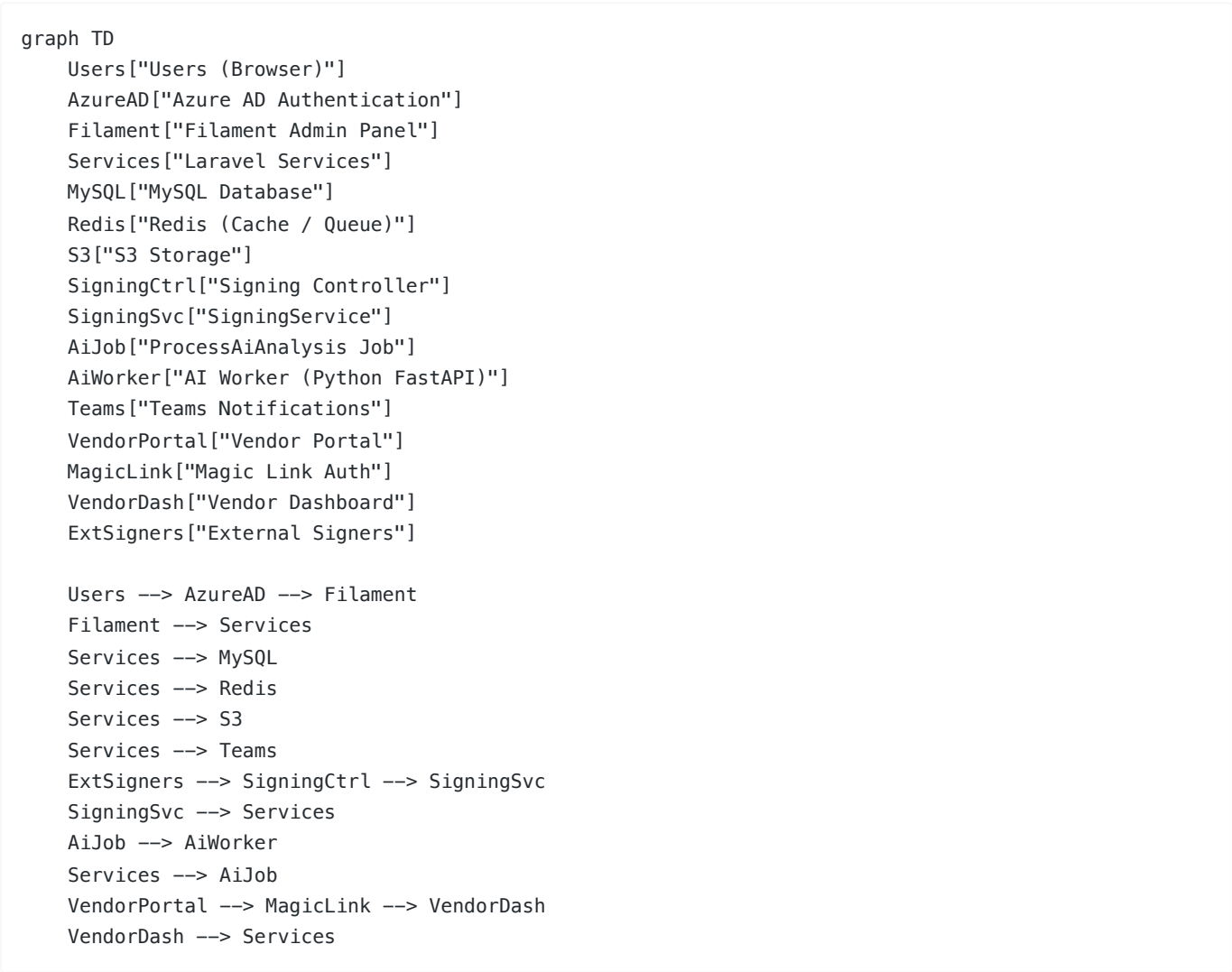
### Key Capabilities

- **Contract Lifecycle Management** -- Create, review, approve, execute, and archive contracts through configurable workflow stages with full audit trails.
- **Electronic Signing (4 capture methods)** -- Collect legally binding signatures via typed text, drawn signature pad, uploaded image, or webcam capture. External signers receive magic-link invitations with no account required.
- **AI-Powered Analysis (5 types)** -- Automated contract review covering risk assessment, clause extraction, compliance checking, obligation identification, and financial term analysis.
- **Compliance Tracking** -- Map contracts to regulatory frameworks, monitor obligations, and generate compliance reports with escalation workflows.

- **Vendor Portal** -- External counterparties log in via magic links to view their agreements, upload requested documents, and respond to KYC requests.
- **Bulk Operations** -- Upload contracts or structured data in bulk via CSV/Excel, with validation, duplicate detection, and progress tracking.

## Platform Architecture

The diagram below shows how CCRS components connect at a high level.



Component	Technology	Purpose
Admin Panel	Filament 3 (Livewire)	All internal user interfaces
Backend	PHP 8.4 / Laravel 12	Business logic, APIs, queue jobs
Database	MySQL 8.0	Persistent storage for all entities
Cache & Queue	Redis 7	Session management, job queue, caching
File Storage	S3-compatible	Contract PDFs, uploaded documents, signatures
AI Worker	Python FastAPI	Contract analysis via Claude SDK
Auth	Azure AD (Socialite)	Single sign-on for internal users

## Navigation Overview

CCRS organises its interface into navigation groups. The sections visible to each user depend on their assigned role.

```
graph LR
    SA["system_admin"]
    LG["legal"]
    CM["commercial"]
    FN["finance"]
    OP["operations"]
    AU["audit"]

    Contracts["Contracts"]
    Counterparties["Counterparties"]
    MerchantAgreements["Merchant Agreements"]
    Workflows["Workflows"]
    KYC["KYC"]
    OrgStructure["Org Structure"]
    Compliance["Compliance"]
    AuditLogs["Audit Logs"]
    Admin["Admin"]
    Dashboard["Dashboard"]
    Reports["Reports"]
    Analytics["Analytics"]
    KeyDates["Key Dates"]
    Reminders["Reminders"]
    Escalations["Escalations"]
    BulkOps["Bulk Operations"]
    Signing["My Signatures"]

    SA --- Contracts
    SA --- Counterparties
    SA --- MerchantAgreements
    SA --- Workflows
    SA --- KYC
    SA --- OrgStructure
    SA --- Compliance
    SA --- AuditLogs
    SA --- Admin
    SA --- Dashboard
    SA --- Reports
    SA --- Analytics
    SA --- KeyDates
    SA --- Reminders
    SA --- Escalations
    SA --- BulkOps
    SA --- Signing

    LG --- Contracts
    LG --- Counterparties
    LG --- KYC
    LG --- Compliance
    LG --- Escalations
    LG --- Reports

    CM --- Contracts
```

CM --- Counterparties

CM --- MerchantAgreements

CM --- KeyDates

CM --- Reminders

FN -. "view only" .-> Contracts

FN --- Reports

FN --- Analytics

OP -. "view only" .-> Contracts

OP --- KeyDates

OP --- Reminders

AU -. "view only" .-> Contracts

AU --- Compliance

AU --- AuditLogs

AU --- Reports

Dashed lines indicate view-only access. Solid lines indicate full read/write access to that navigation group.

Navigation Groups at a Glance

Group	Key Pages	Description
Contracts	Contract list, Merchant Agreements	Core contract records and related merchant agreements
Counterparties	Counterparty list	Organisations and individuals that are parties to contracts
Workflows	Workflow Templates	Configurable approval and review workflow definitions
KYC	KYC Templates	Know-Your-Customer document templates and checklists
Org Structure	Regions, Entities, Projects	Organisational hierarchy used for contract ownership and reporting
Compliance	Regulatory Frameworks, Audit Logs	Regulatory mapping, obligation tracking, and immutable audit trail
Admin	Signing Authorities, Jurisdictions, Vendor Users	System configuration and user/vendor management
Dashboard	Dashboard	Summary widgets showing contract counts, upcoming dates, and alerts
Reports	Reports, Analytics Dashboard, AI Cost Report	Financial and operational reporting with export to Excel/PDF
Key Dates & Reminders	Key Dates, Reminders	Expiry tracking, renewal windows, and configurable reminder schedules
Escalations	Escalations	Overdue and at-risk items requiring management attention
Bulk Operations	Bulk Contract Upload, Bulk Data Upload	Mass import of contracts and structured reference data

Signing	My Signatures	Personal queue of documents awaiting the current user's signature
---------	---------------	---

---

## Roles and Permissions

CCRS uses six predefined roles. Each user is assigned exactly one role, which determines the navigation groups, pages, and actions available to them.

### System Admin

Full access to every area of the platform. System Admins configure the organisational structure (regions, entities, projects), define workflow templates, manage signing authorities and user roles, and perform bulk data uploads. This role is intended for IT administrators and platform owners.

### Legal

Manages the substantive content of contracts. Legal users create and edit contracts, trigger AI-powered analysis, conduct clause-level redlining, manage counterparty records, oversee KYC processes, and monitor compliance against regulatory frameworks. They also handle escalations and generate compliance reports.

### Commercial

Focuses on contract origination and counterparty relationships. Commercial users create new contracts, onboard counterparties, generate merchant agreements from templates, submit override requests for approval thresholds, and track key dates and reminders for their portfolio.

### Finance

Read-only access to contracts with full access to financial reporting. Finance users view contract details without editing, run reports from the analytics dashboard, track AI processing costs, and export financial summaries to Excel and PDF.

### Operations

Day-to-day monitoring of contract timelines. Operations users have read-only access to contract records and focus on key date tracking and reminder management to ensure renewals, expirations, and deliverables are actioned on time.

### Audit

Read-only oversight for governance and compliance. Audit users can view all contract records and their complete audit logs, access compliance reports, and review regulatory framework mappings. They cannot modify any data.

---

## Support

For questions, issues, or feature requests, contact the CCRS support team at [support@digittal.io](mailto:support@digittal.io).

---

## Getting Started

This chapter walks you through your first session with CCRS -- from signing in to finding your way around the interface.

---

## Logging In

CCRS uses **Azure Active Directory (Microsoft) Single Sign-On** for authentication. There are no separate CCRS usernames or passwords to manage.

1. Open your browser and navigate to the CCRS application URL provided by your organisation.

2. You will see the CCRS login page with a **"Sign in with Microsoft"** button.
3. Click the button. You will be redirected to the Microsoft login page.
4. Enter your **corporate Microsoft credentials** (the same email and password you use for Outlook, Teams, etc.) and complete any multi-factor authentication prompts.
5. Once authenticated, Microsoft redirects you back to CCRS and you are logged in.

## First-Time Users

When you sign in for the first time, CCRS automatically provisions your account based on your **Azure AD group membership**. Your Azure AD group determines which CCRS role you are assigned (for example, Legal, Commercial, Finance, Operations, or Audit). You do not need to register or request a separate account.

If you are unable to sign in or you see an "Access Denied" message, contact your **System Administrator** to verify that your Azure AD account is in the correct group.

---

## The Dashboard

After a successful login you land on the **Dashboard** -- your central overview of contract activity across the organisation. The Dashboard is composed of several widgets, each providing a focused summary.

### Contract Status

A breakdown of all contracts grouped by their current workflow state (Draft, Review, Approval, Active, Expired, Terminated, and so on). Use this widget to get a quick picture of where contracts sit in the lifecycle.

### Expiry Horizon

Highlights contracts that are approaching their expiration date within the next **30, 60, and 90 days**. This is your early-warning system for renewals and renegotiations.

### Pending Workflows

Lists action items that are **awaiting your approval or input**. If a workflow stage is assigned to you (or your role), it appears here. Click an item to go directly to the relevant contract.

### Active Escalations

Shows workflow stages that are **overdue** and have been escalated. These require immediate attention to prevent bottlenecks.

### AI Cost

A spending summary for AI-powered contract analysis. Displays token usage and associated costs so the organisation can monitor AI expenditure.

### Compliance Overview

Displays the regulatory compliance status across your contract portfolio. Flags contracts that have outstanding compliance issues or missing regulatory checks.

### Contract Pipeline Funnel

A visual funnel showing how contracts progress through lifecycle stages -- from initial draft through to fully executed. Useful for spotting stages where contracts tend to stall.

### Obligation Tracker

Lists upcoming contract obligations and deadlines (payment milestones, deliverables, reporting requirements, etc.) so nothing falls through the cracks.

## Risk Distribution

Groups contracts by **risk level** as determined by AI analysis (Low, Medium, High, Critical). Helps Legal and Compliance teams prioritise their review workload.

## Workflow Performance

Shows the **average time spent** in each workflow stage. Use this to identify process bottlenecks and measure improvements over time.

---

# Navigation

## Left Sidebar

The **left sidebar** is your primary navigation menu. It is organised by feature area -- Contracts, Counterparties, Workflows, Reports, Administration, and more.

Menu items are **role-based**: you will only see the sections and pages that your assigned role permits. For example, a user with the Finance role will see finance-related reports but may not see system administration pages.

## Global Search

Press **Cmd+K** (macOS) or **Ctrl+K** (Windows/Linux) at any time to open the **Global Search** overlay. From here you can quickly search across:

- Contracts (by title, reference number, or content)
- Counterparties (by name or registration number)
- Other records throughout the system

Start typing and results appear instantly. Click a result to navigate directly to that record.

## Breadcrumbs

Every page displays a **breadcrumb trail** at the top of the content area. Breadcrumbs show your current location in the application hierarchy and let you jump back to any parent page with a single click.

## Notifications Bell

In the **top-right corner** of every page you will find the notifications bell icon. A badge indicates the number of **unread notifications**. Click the bell to expand the notifications panel and review recent alerts such as workflow assignments, escalation notices, and contract status changes.

---

# Profile and Preferences

Click your **name or avatar** in the top-right corner of the page to access your profile menu.

## Notification Preferences

From the profile menu, navigate to **Notification Preferences** to control how and when CCRS contacts you. You can configure each notification type independently across the following channels:

- **Email** -- notifications sent to your corporate email address
- **Microsoft Teams** -- notifications posted via the Teams integration
- **In-App** -- notifications shown in the CCRS notifications panel (the bell icon)
- **Calendar ICS** -- calendar invitations sent for deadline-based notifications (e.g., contract expiry reminders, obligation due dates)

Adjust these settings to match your workflow. For example, you might choose to receive escalation alerts via both email and Teams, but limit routine status updates to in-app only.

---

## Getting Help

If you need assistance while using CCRS:

1. **Help and Guide page** -- accessible from the left sidebar (look for the question mark icon). This page contains contextual guidance and frequently asked questions.
  2. **Support** -- for issues not covered by the in-app guide, or if you encounter a technical problem, contact [support@digittal.io](mailto:support@digittal.io).
- 

## 3. Contract Lifecycle

### Overview

Every contract in CCRS moves through a defined series of workflow states from creation to completion. The lifecycle ensures proper review, approval, and signing before a contract becomes legally binding. There are seven workflow states:

State	Description	Key Participants
<b>Draft</b>	The contract has been created but is not yet ready for review. Authors can edit all fields, upload or replace files, and run AI analysis.	Contract Author
<b>Review</b>	The contract has been submitted for review. Reviewers examine the terms, run AI analysis, and either approve the review or request changes.	Assigned Reviewers
<b>Approval</b>	The reviewed contract is awaiting formal approval from an authorized approver. The approver can approve it to proceed to signing or reject it back to review.	Approvers (per workflow template)
<b>Signing</b>	The contract has been approved and is awaiting the external party's signature. A signing invitation is sent to the counterparty.	Contract Author, External Counterparty
<b>Countersign</b>	The external party has signed. The contract now requires an internal countersignature to finalize the agreement.	Internal Countersigner
<b>Executed</b>	Both parties have signed. The contract is now legally binding and <b>immutable</b> -- no fields can be edited. Amendments, renewals, and side letters can be created from this state.	All stakeholders (read-only)
<b>Archived</b>	The contract term has ended, or it has been superseded by a renewal or amendment. The contract is permanently read-only.	All stakeholders (read-only)

A contract can also be **Cancelled** from any active state (Draft, Review, Approval, or Signing).

### Contract Lifecycle State Machine

```
stateDiagram-v2
    [*] --> Draft
    Draft --> Review : Submit for Review
    Review --> Approval : Approve Review
    Review --> Draft : Request Changes
    Approval --> Signing : Approved
    Approval --> Review : Reject
    Signing --> Countersign : External Party Signed
    Countersign --> Executed : Countersigned
    Executed --> Archived : Term Ended / Replaced
```

```

Draft --> Cancelled : Cancel
Review --> Cancelled : Cancel
Approval --> Cancelled : Cancel
Signing --> Cancelled : Cancel

state Executed {
    [*] --> ReadOnly
    note right of ReadOnly : Immutable – no edits allowed
}

state Archived {
    [*] --> Permanent
    note right of Permanent : Read-only archive
}

```

## Creating a Contract

Follow the steps below to create a new contract. The creation flow ensures that every contract is associated with the correct organizational context and counterparty before any content is added.

### Contract Creation Flow

```

flowchart TD
    A[User clicks New Contract] --> B[Select Region]
    B --> C[Select Entity]
    C --> D[Select Project]
    D --> E[Select Counterparty]
    E --> F{Contract Type?}
    F -->|Commercial| G[Enter Details + Upload File]
    F -->|Merchant| H[Generate from Template]
    G --> I[Save Contract]
    H --> I
    I --> J[Contract Created in Draft State]
    J --> K{Matching Workflow Template?}
    K -->|Yes| L[Workflow Auto-Assigned]
    K -->|No| M[Stays in Draft Until Template Created]
    L --> N[Ready for Workflow Progression]
    M --> N

```

### Step-by-Step

1. **Click "New Contract"** from the Contracts list page.
2. **Select Region** -- Choose the geographic region this contract belongs to. Regions are configured by your administrator.
3. **Select Entity** -- Choose the legal entity entering into the agreement (filtered by the selected region).
4. **Select Project** -- Choose the project this contract is associated with (filtered by the selected entity).
5. **Select Counterparty** -- Choose the external party you are contracting with. If the counterparty does not exist, you can create one from this screen.
6. **Choose Contract Type** -- Select either *Commercial* or *Merchant* (see Contract Types below).
7. **Fill in contract details:**
  - **Title** -- A descriptive name for the contract.
  - **Description** -- A summary of the contract's purpose and key terms.
  - **Start Date** -- The date the contract takes effect.
  - **End Date** -- The date the contract expires.
  - **Value** -- The monetary value of the contract.
  - **Currency** -- The currency for the contract value.

- **File Upload** -- Upload the contract document (PDF or DOCX) for Commercial contracts, or generate from a template for Merchant contracts.

8. **Save** -- The contract is created in **Draft** state.

9. **Workflow Assignment** -- If a WorkflowTemplate exists that matches the contract type and region/entity/project combination, it is automatically assigned. If no matching template exists, the contract remains in Draft until an administrator creates a matching workflow template.

---

## Contract Types

### Commercial

Commercial contracts are standard agreements uploaded as PDF or DOCX files. These cover a wide range of business arrangements including vendor agreements, service contracts, partnership agreements, and any other commercial arrangement.

- Upload an existing contract document (PDF or DOCX format).
- All contract metadata (title, dates, value, etc.) is entered manually.
- AI analysis can be run on uploaded documents to extract key terms and flag potential issues.

### Merchant

Merchant contracts are generated from a master template (WikiContract) using the Merchant Agreement generator. This ensures consistency across all merchant agreements and reduces manual drafting effort.

- Select a WikiContract template during creation.
- Template fields are pre-populated based on the selected counterparty and organizational context.
- The generated document can be reviewed and adjusted before submission.

---

## File Management

CCRS supports secure document storage and external document management integration.

- **Supported formats:** PDF and DOCX files can be uploaded for any contract.
- **Secure storage:** Uploaded files are stored in S3 with access controlled by CCRS permissions. Only users with appropriate access to the contract can download files.
- **File replacement:** While a contract is in Draft state, the uploaded file can be replaced with an updated version.
- **SharePoint integration:** For organizations using SharePoint for document management, each contract record supports:
  - **SharePoint URL** -- A link to the document in your SharePoint library for quick external access.
  - **SharePoint Version** -- Tracks which version of the SharePoint document corresponds to the CCRS record.

---

## Contract Actions by State

The actions available on a contract depend on its current workflow state. The table below summarizes what you can do at each stage.

### Draft

Action	Description
Edit	Modify any contract field (title, dates, value, description, etc.)
Upload new file	Upload or replace the contract document (PDF/DOCX)
Delete	Permanently remove the contract
Send for Review	Submit the contract to move it into the Review state
AI Analysis	Run AI-powered analysis to extract key terms, flag risks, and summarize the document

## Review

Action	Description
AI Analysis	Run or re-run AI analysis on the contract document
Approve	Approve the review and advance the contract to the Approval state
Request Changes	Return the contract to Draft with reviewer comments for the author to address
Cancel	Cancel the contract entirely

## Approval

Action	Description
Approve	Formally approve the contract and advance it to the Signing state
Reject	Return the contract to the Review state for further examination
Cancel	Cancel the contract entirely

## Signing

Action	Description
Send Signing Invitation	Send a digital signing request to the external counterparty
Cancel	Cancel the contract entirely

## Countersign

Action	Description
Countersign	Apply the internal countersignature to finalize the contract

## Executed

Action	Description
Download	Download the executed contract document
Create Amendment	Create a linked amendment to modify terms of this contract
Create Renewal	Create a linked renewal to extend or replace this contract
Create Side Letter	Create a linked supplementary agreement

All other fields are **read-only**. The contract cannot be edited once executed.

## Archived

Action	Description
Download	Download the archived contract document

Archived contracts are **fully read-only**. No modifications of any kind are permitted.

---

## Access Control

### Restricted Contracts

Contracts can be marked as **restricted** to limit visibility to a specific set of authorized users. This is useful for sensitive agreements such as executive compensation, M&A transactions, or confidential vendor arrangements.

- **Who can restrict:** System Admins and users with the Legal role can mark a contract as restricted or unrestrict it.
  - **Effect of restriction:** Once restricted, the contract is hidden from all users except:
    - **System Admins** -- always have access to all contracts.
    - **Authorized users** -- users explicitly added to the contract's access list.
  - **Managing access:** The authorized user list is managed through the ContractUserAccess table. Admins and Legal users can add or remove users from this list via the contract detail page.
  - **Unrestricting:** Removing the restriction makes the contract visible to all users with standard permissions again.
- 

### Linked Contracts

Contracts in CCRS can be linked to other contracts to represent formal relationships between agreements. All linked contracts are managed through the ContractLink model and are created from an **Executed** contract's action menu.

#### Amendments

Amendments are formal modifications to an existing executed contract. They are used when specific terms, conditions, or values need to change without replacing the entire agreement.

- Created from the parent contract's action menu.
- Linked to the parent contract with the relationship type `amendment`.
- The amendment follows its own lifecycle (Draft through Executed) independently.
- The parent contract remains unchanged and read-only.

#### Renewals

Renewals represent the extension or replacement of an existing contract. When a contract term is ending, a renewal creates a new contract linked to the original.

- Created from the original contract's action menu.
- Linked to the original contract with the relationship type `renewal`.
- Once the renewal is executed, the original contract can be archived (Term Ended / Replaced).
- The renewal carries forward context from the original contract.

#### Side Letters

Side letters are supplementary agreements that accompany and modify aspects of a parent contract without formally amending it. They are commonly used for clarifications, temporary modifications, or party-specific arrangements.

- Created from the parent contract's action menu.
  - Linked to the parent contract with the relationship type `side_letter`.
  - Each side letter follows its own lifecycle independently.
  - Multiple side letters can be linked to a single parent contract.
- 

## Immutability

Once a contract reaches the **Executed** or **Archived** state, it is locked for compliance purposes. This means:

- All fields become **read-only** and cannot be modified through the UI or API.
- The uploaded contract document cannot be replaced or deleted.
- Workflow state changes are limited (Executed can move to Archived; Archived is permanent).

- Audit trails for the contract are preserved and cannot be altered.

To make changes to the terms of an executed contract, you must create one of the following linked contracts:

- **Amendment** -- for modifying specific terms or conditions.
- **Renewal** -- for extending or replacing the contract.
- **Side Letter** -- for supplementary clarifications or temporary modifications.

Each linked contract goes through its own full lifecycle, ensuring that all changes are properly reviewed, approved, and signed before taking effect.

---

## Counterparty Management

Counterparties are the organisations and individuals that are parties to your contracts. CCRS provides a complete set of tools for onboarding counterparties, detecting duplicates, managing compliance status, and maintaining contact information -- all from a single interface.

---

### Creating a Counterparty

To add a new counterparty to the system:

1. Navigate to **Counterparties** in the left sidebar.
2. Click the **"New Counterparty"** button in the top-right corner of the list page.
3. Fill in the required fields:
  - **Legal Name** -- the full legal name of the organisation or individual.
  - **Registration Number** -- the company registration or incorporation number.
4. Optionally complete the remaining fields:
  - **Jurisdiction** -- the legal jurisdiction in which the counterparty is registered.
  - **Description** -- any additional context or notes about the counterparty.
5. **Before saving**, click the **"Check for Duplicates"** button to verify that this counterparty does not already exist in the system (see the next section for details).
6. Once you are satisfied there is no duplicate, click **Save**. The counterparty is created with a status of **Active**.

### Who Can Create Counterparties?

Users with the **System Admin**, **Legal**, or **Commercial** role can create new counterparty records.

### Onboarding Flow

The diagram below shows the full onboarding process, including duplicate checking and KYC assignment.

```
flowchart TD
    A[Navigate to Counterparties] --> B[Click 'New Counterparty']
    B --> C[Enter Legal Name & Registration Number]
    C --> D[Click 'Check for Duplicates']
    D --> E{Duplicates Found?}
    E -->|Yes| F[Review Matches]
    F --> G{Proceed Anyway?}
    G -->|No| H[Use Existing Record]
    G -->|Yes| I[Acknowledge & Continue]
    E -->|No| I
    I --> J[Complete Remaining Fields]
    J --> K[Save - Status: Active]
    K --> L{KYC Template Assigned?}
    L -->|Yes| M[Complete KYC Checklist Items]
```

```
L -->|No| N[Ready for Contracts]
M --> N
```

## Duplicate Detection

Duplicate counterparty records create confusion and split contract history across multiple entries. CCRS includes built-in duplicate detection to prevent this.

### How It Works

When you click the "**Check for Duplicates**" button on the counterparty creation form, CCRS performs a server-side search that:

- **Fuzzy-matches the Legal Name** -- finds counterparties with similar (not necessarily identical) names, accounting for minor spelling variations, abbreviations, and word order differences.
- **Exact-matches the Registration Number** -- finds any existing counterparty with the same registration number, which is a strong indicator of a true duplicate.

### Reviewing Results

If potential duplicates are found, a modal dialog appears showing:

- The **matching counterparty records** with their legal name, registration number, jurisdiction, and current status.
- A **match confidence indicator** so you can assess how likely it is that the match is a true duplicate.

You then have two choices:

1. **Use the existing record** -- close the form and navigate to the matching counterparty instead. This is the recommended action when a true duplicate is detected.
2. **Acknowledge and continue** -- if you have determined that the match is a false positive (for example, two distinct companies with similar names), you can acknowledge the finding and proceed with creating the new record.

## Status Management

Every counterparty has one of three statuses that controls whether new contracts can be created against them.

Status	Meaning	Contract Creation
Active	The counterparty is in good standing.	Allowed -- no restrictions.
Suspended	A temporary restriction has been placed on the counterparty.	Blocked -- requires an approved <b>Override Request</b> .
Blacklisted	A permanent restriction has been placed on the counterparty.	Blocked -- requires an approved <b>Override Request</b> .

### Changing Status

**System Admin** and **Legal** users can change a counterparty's status from the counterparty edit page. When a counterparty is moved to Suspended or Blacklisted, any attempt to associate them with a new contract will be blocked until an override is granted.

## Override Requests

When a Commercial user needs to create a contract with a **Suspended** or **Blacklisted** counterparty, they must submit an Override Request for approval.

### Submitting a Request

1. When you attempt to use a restricted counterparty, the system will prompt you to submit an override request.
2. Provide a **business justification** -- a clear explanation of why the restriction should be waived for this particular contract.
3. The request is created with a status of **Pending** and routed to Legal and System Admin users for review.

## Review and Decision

A **Legal** or **System Admin** user reviews the override request and takes one of two actions:

- **Approve** -- the override is granted and the Commercial user can proceed with creating the contract against the restricted counterparty.
- **Reject** -- the override is denied. The reviewer provides a **comment** explaining the reason for rejection.

The requesting user is notified of the outcome.

## Approval Flow

```
flowchart LR
    A[Commercial User] --> B[Counterparty is Suspended/Blacklisted]
    B --> C[Submit Override Request]
    C --> D[Provide Business Justification]
    D --> E[Request Status: Pending]
    E --> F[Legal / Admin Reviews]
    F -->|Approve| G[Override Granted – Proceed with Contract]
    F -->|Reject| H[Override Denied – Comment Provided]
```

## Merging Counterparties (Admin Only)

Over time, duplicate counterparty records may be discovered -- for example, the same company entered under slightly different names. System Admins can merge these records to consolidate contract history.

### How to Merge

1. Open the **source** counterparty (the duplicate record you want to eliminate).
2. Select the **"Merge Into"** action.
3. Search for and select the **target** counterparty (the record you want to keep).
4. Review both records displayed **side-by-side** to confirm the merge is correct.
5. Click **Confirm** to execute the merge.

### What Happens During a Merge

- All **contracts** associated with the source counterparty are transferred to the target counterparty.
- The source counterparty's status is changed to **Merged**, with a reference to the target record.
- An **audit log entry** is created recording the merge, including who performed it and when.

### Important

Merging is **irreversible**. Once a merge is confirmed, the source counterparty cannot be restored to its original state. Always verify both records carefully before confirming.

## Merge Flow

```
flowchart TD
    A[System Admin Identifies Duplicate] --> B[Open Source Counterparty]
    B --> C[Select 'Merge Into' Action]
    C --> D[Search & Select Target Counterparty]
    D --> E[Review: Both Records Shown Side-by-Side]
```

```
E --> F[Confirm Merge]
F --> G[All Contracts Moved to Target]
G --> H[Source Counterparty Marked 'Merged']
H --> I[Audit Log Entry Created]
```

---

## Contacts

Each counterparty can have multiple **contacts** -- the people you interact with at that organisation. Contacts are managed from the counterparty's edit page.

### Contact Fields

Field	Description
<b>Name</b>	The contact's full name.
<b>Email</b>	The contact's email address.
<b>Phone</b>	The contact's phone number.
<b>Position</b>	The contact's role or job title at the counterparty organisation.

### Managing Contacts

1. Open the counterparty record and navigate to the **Contacts** tab (relation manager).
2. Click **"Add Contact"** to create a new contact entry.
3. Fill in the contact details and save.
4. To edit or remove an existing contact, use the action buttons on the contact row.

Contacts are used throughout the system -- for example, when setting up signing sessions or sending notifications related to a contract.

---

## Stored Signatures

CCRS allows administrators to manage **stored signatures** for counterparty representatives. These pre-captured signatures can be applied during contract signing sessions, streamlining the execution process.

### Who Can Manage Stored Signatures?

Only **System Admin** and **Legal** users can create, edit, or delete stored signatures for a counterparty.

### Managing Stored Signatures

1. Open the counterparty record and navigate to the **"Stored Signatures"** tab.
  2. From this tab you can add new signatures, view existing ones, or remove signatures that are no longer valid.
  3. When a signing session is initiated for a contract involving this counterparty, the stored signatures are available for selection.
- 

## KYC Compliance

Know Your Customer (KYC) compliance ensures that adequate due diligence has been performed on a counterparty before entering into contracts with them.

### KYC Templates

A **KYC Template** defines a checklist of items required for counterparty due diligence. Templates are configured by System Admins and typically include items such as:

- Proof of incorporation or registration
- Identification documents for directors or beneficial owners
- Financial statements or references
- Sanctions screening results

## KYC Packs

When a KYC Template is **assigned to a counterparty**, CCRS creates a **KYC Pack** -- an instance of that template linked to the specific counterparty. The KYC Pack contains individual checklist items that must be completed.

## Completing KYC

1. **Legal** users review the KYC Pack on the counterparty's record.
2. Each checklist item is marked as **complete** once the required documentation or verification has been obtained.
3. Progress is tracked at the pack level so you can see at a glance how many items are outstanding.
4. Once all items are complete, the counterparty's KYC status is satisfied and they are fully onboarded for contract activity.

## Summary of Role Permissions

The table below summarises which roles can perform key counterparty management actions.

Action	System Admin	Legal	Commercial	Finance	Operations	Audit
Create counterparty	Yes	Yes	Yes	--	--	--
Edit counterparty	Yes	Yes	--	--	--	--
Change status	Yes	Yes	--	--	--	--
Merge counterparties	Yes	--	--	--	--	--
Submit override request	--	--	Yes	--	--	--
Review override request	Yes	Yes	--	--	--	--
Manage stored signatures	Yes	Yes	--	--	--	--
Complete KYC checklist	--	Yes	--	--	--	--
View counterparty records	Yes	Yes	Yes	--	--	--

## 5. Electronic Signing

### Overview

CCRS includes a full in-house electronic signing system. No external signing providers (DocuSign, Adobe Sign, etc.) are required. The system supports sequential and parallel signing orders, four signature capture methods, stored signatures for repeat use, template-based signing blocks, page enforcement controls, and a complete audit trail with document integrity verification.

Every signing action -- creating a session, viewing the document, signing, declining, sending reminders -- is recorded in the audit log with the actor's IP address and user agent string.

### Creating a Signing Session

When a contract has been approved and is ready for execution, the initiator creates a signing session from the contract's action menu.

1. Navigate to the contract and click **Send for Signing** in the action menu.
2. Choose the **signing order**:
  - **Sequential** -- signers receive their invitations one at a time, in the order you specify.
  - **Parallel** -- all signers receive their invitations simultaneously.
3. Add signers. For each signer, provide:
  - **Name** -- the signer's full name.
  - **Email** -- the email address where the invitation will be sent.
  - **Type** -- Internal (a user within your organisation) or External (a counterparty or third party).
  - **Order** -- the signing position (used in sequential mode to determine who signs first, second, etc.).
4. Configure optional enforcement settings:
  - **Require all pages viewed** -- signers must scroll through every page of the PDF before the submit button becomes available.
  - **Require page initials** -- signers must initial each page of the document before they can submit their final signature.
5. Review the session details and activate it.

If the contract was generated from a WikiContract template that has pre-defined signing blocks, the signature field positions are auto-populated -- you do not need to place them manually.

The session is valid for **30 days** from creation. Individual signer tokens expire after **7 days**, after which the initiator can resend the invitation.

---

## Sequential vs Parallel Signing

### Sequential Signing

In sequential mode, each signer receives their email invitation only after the previous signer in the order has completed their signature. This is the appropriate choice when a specific signing order is required -- for example, when the counterparty must sign before the company countersigns.

flowchart TD

```
A[Initiator Creates Signing Session] --> B[Set Order: Sequential]
B --> C[Add Signers with Order Numbers]
C --> D[Session Activated]
D --> E[Email Sent to Signer 1]
E --> F[Signer 1 Opens Link]
F --> G[Views PDF Document]
G --> H{Page Enforcement?}
H -->|Yes| I[Must View All Pages]
H -->|No| J[Review Document]
I --> J
J --> K[Choose Signature Method]
K --> L[Submit Signature]
L --> M[advanceSession Called]
M --> N{More Signers?}
N -->|Yes| O[Email Sent to Next Signer]
O --> F
N -->|No| P[completeSession Called]
P --> Q[Signatures Overlaid on PDF]
Q --> R[Audit Certificate Generated]
R --> S[Final Document Hash Computed]
S --> T[Completion Emails Sent]
T --> U[Contract Marked as Signed]
```

## Parallel Signing

In parallel mode, all signers receive their email invitations at the same time and can sign in any order. The session completes automatically once every signer has submitted their signature. This mode is faster when no particular signing order is required.

```
flowchart TD
    A[Initiator Creates Signing Session] --> B[Set Order: Parallel]
    B --> C[Add Signers]
    C --> D[Session Activated]
    D --> E[Emails Sent to ALL Signers Simultaneously]
    E --> F[Each Signer Signs Independently]
    F --> G[advanceSession Checks Progress]
    G --> H{All Signed?}
    H -->|No| I[Wait for Remaining Signers]
    I --> F
    H -->|Yes| J[completeSession Called]
    J --> K[Signatures Overlaid on PDF]
    K --> L[Audit Certificate + Final Hash]
    L --> M[Completion Emails + Contract Marked Signed]
```

## The Signer's Experience

External signers do not need a CCRS account. They receive a magic-link email invitation and interact with the signing page directly. The step-by-step experience is as follows:

1. **Receive email** -- the signer receives an email containing a signing invitation link. The link includes a unique, cryptographically generated token.
2. **Open the signing page** -- clicking the link opens the CCRS signing page in the browser. No login is required.
3. **View the contract PDF** -- the full contract document is displayed in an embedded PDF viewer.
4. **Scroll through pages (if required)** -- if page enforcement is enabled, a progress bar shows "Pages viewed: X/Y". The signer must scroll through every page before the submit button becomes active.
5. **Initial each page (if required)** -- if page initials are required, each page displays an "Initial" button. Clicking it opens a mini-canvas where the signer draws their initials (or selects previously stored initials). The progress indicator shows "Pages initialed: X/Y".
6. **Choose a signature method** -- the signer selects one of four capture methods: Draw, Type, Upload, or Camera. If the signer has previously stored signatures in CCRS, those appear for one-click selection.
7. **Review and submit** -- the signer reviews their signature and clicks Submit.
8. **Save for future use (optional)** -- after signing, a "Save this signature for future use" checkbox is offered. Checking it stores the signature for faster signing next time.
9. **Confirmation** -- a confirmation message is displayed, and the signer receives a confirmation email.

## Page Enforcement in Detail

Page enforcement gives the initiator control over how thoroughly signers must review the document before signing.

```
flowchart TD
    A[PDF Document Renders in Viewer] --> B[Intersection Observer Tracks Pages]
    B --> C[Progress Bar: Pages Viewed X/Total]
    C --> D{require_all_pages_viewed?}
    D -->|Yes| E[User Must Scroll Through Every Page]
    D -->|No| F[Proceed to Signing]
    E --> G{All Pages Viewed?}
    G -->|No| H[Submit Button Disabled]
    G -->|Yes| I{require_page_initials?}
    I -->|Yes| J[Each Page Shows 'Initial' Button]
```

```
J --> K[Click → Mini-Canvas Opens]
K --> L[Draw Initials or Use Stored Initials]
L --> M[Mark Page as Initialed]
M --> N{All Pages Initialed?}
N -->|No| H
N -->|Yes| F
I -->|No| F
F --> O[Submit Button Enabled]
```

- **Viewing requirement** -- when `require_all_pages_viewed` is enabled, the system uses a browser Intersection Observer to track which pages the signer has scrolled past. The submit button remains disabled until every page has been seen.
- **Page initials** -- when `require_page_initials` is enabled, each page of the PDF displays an "Initial" button. Clicking the button opens a mini-canvas overlay where the signer draws their initials. If the signer has stored initials marked as default, those are applied automatically with a single click.
- **Progress indicators** -- the signing page displays real-time progress: "Pages viewed: 3/12" and "Pages initialed: 3/12".

---

## Declining to Sign

A signer can choose to decline rather than sign. The process is as follows:

1. On the signing page, click **Decline**.
2. Provide a **reason** for declining (required).
3. The decline is recorded in the audit log with the signer's IP address and user agent.
4. The session initiator receives an email notification that the signer has declined, along with the stated reason.
5. The signer's status is updated to "declined". A declined signer cannot later sign or re-decline.

Depending on the session configuration, a decline may halt the entire signing process (in sequential mode, subsequent signers will not receive their invitations).

---

## Signature Capture Methods

CCRS supports four methods for capturing signatures. All four produce a PNG image that is overlaid on the PDF at the designated signing position.

### Draw

The signer draws their signature on an HTML canvas using a mouse, trackpad, or touchscreen. This is the most common method and works on all devices.

- A clear button allows the signer to start over.
- The canvas captures smooth strokes in real time.

### Type

The signer types their name, and CCRS renders it as a signature-style image using a script font. This method is the fastest and most convenient when a handwritten appearance is not critical.

### Upload

The signer uploads an existing PNG or JPEG file containing their signature. This is useful for signers who have a pre-prepared signature image -- for example, a scan of their handwritten signature.

### Camera / Webcam

The signer uses their device's camera to photograph a handwritten signature on paper. CCRS processes the captured image to produce a clean, transparent-background signature.

```

flowchart LR
    A[Click Camera Tab] --> B[Browser Requests Camera Access]
    B --> C[Live Video Preview Shown]
    C --> D[Hold Paper with Signature to Camera]
    D --> E[Click Capture Button]
    E --> F[Frame Grabbed from Video]
    F --> G[Convert to Grayscale]
    G --> H[Apply Threshold Filter]
    H --> I[Remove Light Background → Transparent]
    I --> J[Preview Processed Image]
    J --> K{Accept?}
    K -->|No| C
    K -->|Yes| L[Signature Ready for Use]

```

The image processing pipeline works as follows:

1. A single frame is captured from the video feed.
2. The image is converted to grayscale.
3. A threshold filter is applied to isolate the dark ink of the signature from the lighter background of the paper.
4. Light pixels (the paper background) are made transparent, leaving only the signature strokes.
5. The processed image is previewed. The signer can accept it or re-capture.

## Stored Signatures

CCRS allows users and signers to save their signatures for future use, eliminating the need to re-draw or re-capture each time.

### Managing Stored Signatures

The **My Signatures** page is accessible from the user menu. From this page, users can:

- **Add a new signature or initials** using any of the four capture methods (Draw, Type, Upload, Camera).
- **Set a default** -- mark one signature and one initials as the default. Defaults are automatically pre-selected during signing.
- **Delete** stored signatures that are no longer needed.

```

flowchart TD
    A[User Opens 'My Signatures' Page] --> B{Choose Action}
    B -->|Add New| C[Select Capture Method]
    C --> D{Method?}
    D -->|Draw| E[Draw on Canvas]
    D -->|Type| F[Type Name → Rendered as Image]
    D -->|Upload| G[Upload PNG/JPEG File]
    D -->|Webcam| H[Camera Capture]
    E --> I[Enter Label & Choose Type]
    F --> I
    G --> I
    H --> I
    I --> J[Save to S3 Storage]
    J --> K[Signature Available for Future Signing]
    B -->|Set Default| L[Mark as Default Signature/Initials]
    B -->|Delete| M[Remove Stored Signature]

```

### Using Stored Signatures During Signing

When a signer opens a signing page and has stored signatures on file, the following behaviour applies:

- Stored signatures are displayed in a selection panel alongside the capture method tabs.
- The default signature (if set) is pre-selected.
- The signer can click a stored signature to use it immediately, or switch to any capture method to create a new one.

## Save for Future Use

After completing a signature on a signing page, the signer is offered a **"Save this signature for future use"** checkbox. Checking it creates a new stored signature record linked to the signer's email address (for external signers) or user account (for internal signers).

---

## Template Signing Blocks

When contracts are generated from WikiContract templates, the template can pre-define the positions and types of signing fields on the document.

### How Template Fields Work

Each template signing field specifies:

Property	Description
Field type	Signature, Initials, Text, or Date
Signer role	Who fills this field: company, counterparty, witness_1, witness_2, etc.
Page number	Which page of the PDF the field appears on
Position	X/Y coordinates and width/height for the field placement
Required	Whether the field must be filled before submission

### Role-to-Signer Mapping

When a signing session is created from a template-based contract, CCRS automatically maps template roles to signers:

Template Role	Maps To
company	Internal signer (the company's authorised signatory)
counterparty	External signer (the other party to the agreement)
witness_1	First additional signer (typically a witness)
witness_2	Second additional signer
witness_3	Third additional signer

This mapping means the initiator does not need to manually place signing fields -- they are inherited from the template and automatically assigned to the correct signers.

### Non-Template Contracts

For contracts that were not generated from a template (e.g., uploaded PDFs), the initiator can manually add signing fields during session creation, specifying the field type, page, position, and which signer is responsible for each field.

---

## Session Completion

When all signers have submitted their signatures, the system automatically finalises the signing session. The completion process involves five steps:

1. **Signature overlay** -- all collected signatures are overlaid onto the original PDF document at their designated positions (as defined by template fields or manual placement).
2. **Audit certificate** -- a separate PDF document is generated summarising the signing session: who signed, when they signed, their IP addresses, the signature method used, and the document hash.
3. **Final document hash** -- a SHA-256 hash of the completed, signed PDF is computed and stored. This hash can be used to verify that the document has not been altered after signing.
4. **Notification emails** -- completion emails are sent to all signers and the session initiator, confirming that signing is complete.
5. **Contract status update** -- the contract's signing status is updated to "signed", and the contract advances to the next lifecycle state.

The final signed PDF and audit certificate are stored securely. The original unsigned document is preserved alongside the signed version for reference.

---

## Reminders

If a signer has not acted on their invitation, the session initiator can send a reminder:

- Navigate to the signing session and click **Send Reminder** next to the pending signer.
  - A new email is sent to the signer with the same signing link.
  - The reminder is recorded in the audit log.
  - Reminders do not reset the token expiry. If the 7-day token has expired, a new invitation must be sent instead.
- 

## Session Cancellation

The session initiator can cancel an active signing session at any time:

- Navigate to the signing session and click **Cancel Session**.
  - All pending signer tokens are invalidated.
  - Signers who have not yet signed can no longer access the signing page.
  - Signatures already collected are preserved in the audit record but are not applied to the contract.
  - The contract returns to its previous workflow state.
- 

## Security

The signing system is designed with multiple layers of security to ensure document integrity and signer authenticity.

### Token Security

- **CSPRNG tokens** -- signing invitation tokens are generated using a cryptographically secure pseudo-random number generator.
- **SHA-256 hashed storage** -- only the SHA-256 hash of each token is stored in the database. The plaintext token appears only in the email link. This means that even if the database is compromised, tokens cannot be extracted.
- **Token expiry** -- individual signer tokens expire after **7 days**. The overall session expires after **30 days**.
- **Single use** -- once a signer has signed or declined, the token cannot be reused for a different action.

### Document Integrity

- **Hash at creation** -- a SHA-256 hash of the original document is computed when the signing session is created.
- **Hash at completion** -- a SHA-256 hash of the final signed document is computed and stored.
- **Tamper detection** -- the creation hash can be compared against the original file to verify that the document presented to signers was not altered during the signing process.

### Audit Trail

Every significant action is logged in the signing audit log:

Event	Details Captured
Session created	Initiator, contract, signing order, settings
Invitation sent	Signer email, timestamp
Document viewed	Signer, timestamp, IP address, user agent
Signature submitted	Signer, method, timestamp, IP address, user agent
Signing declined	Signer, reason, timestamp, IP address, user agent
Session completed	Final document hash, timestamp
Session cancelled	Cancelled by, reason, timestamp
Reminder sent	Signer email, timestamp

### Signature Storage

- Signature images are stored securely in S3-compatible object storage.
- Stored signatures are accessible only to the owning user.
- Signature images are not publicly accessible -- they are served through authenticated routes.

## 6. Workflow Templates

### Overview

Workflow templates define the sequence of approval stages a contract moves through from submission to execution. Each template is a reusable blueprint that specifies:

- **Which stages apply** -- Review, Legal Approval, Finance Sign-off, Executive Approval, etc.
- **Who is responsible at each stage** -- defined by role (e.g. Legal, Finance, Operations).
- **How long each stage should take** -- an SLA expressed in days.
- **Whether approval is required** -- some stages may be informational; others require explicit sign-off before the contract can proceed.
- **What happens when SLAs are breached** -- escalation rules that notify progressively senior stakeholders.

Templates are created and managed exclusively by **System Admin** users. Once published, a template automatically assigns itself to new contracts that match its contract type and organizational scope.

### Workflow Approval Flow with Escalation

The diagram below illustrates how a contract moves through a single workflow stage, including the escalation path when an approver does not act within the SLA window.

```

flowchart TD
    A[Contract Enters Workflow Stage] --> B[Assigned Approver Notified]
    B --> C{Approver Action?}
    C -->|Approve| D[Move to Next Stage]
    C -->|Reject| E[Return to Previous Stage]
    C -->|No Action| F[SLA Timer Running]
    F --> G{SLA Breached?}
    G -->|No| C
    G -->|Yes| H[Escalation Tier 1]
    H --> I[Additional Stakeholders Notified]
  
```

```
I --> J{Action Taken?}
J -->|Yes| C
J -->|No| K{Tier 2 Threshold?}
K -->|Yes| L[Escalation Tier 2]
L --> M[Senior Management Notified]
M --> N{Action Taken?}
N -->|Yes| C
N -->|No| O{Tier 3 Threshold?}
O -->|Yes| P[Escalation Tier 3]
P --> Q[Executive Escalation]
D --> R{More Stages?}
R -->|Yes| A
R -->|No| S[Workflow Complete]
```

## Creating a Workflow Template (Admin Only)

Only users with the **System Admin** role can create, edit, publish, or delete workflow templates. Follow these steps to create a new template.

### Step-by-Step

- Navigate to Workflows** -- In the left sidebar, expand the Workflows section and click **Workflow Templates**.
- Click "New"** -- Use the "New" button in the top-right corner of the list page.
- Enter template name and description** -- Choose a clear, descriptive name (e.g. "Commercial Contract -- EMEA Standard Approval") and a summary of the template's purpose.
- Select contract type** -- Choose either *Commercial* or *Merchant*. This determines which contracts the template can be assigned to.
- Optionally scope to a Region, Entity, or Project** -- Select one or more of these to restrict the template to a specific part of the organization. Leave all three blank to create a global template that applies to any contract of the selected type.
- Build the workflow stages** -- Use the Visual Workflow Builder (drag-and-drop) or the AI Generation feature to define the sequence of approval stages. See the sections below for details on each approach.
- Configure escalation rules** -- For each stage, define what happens when the SLA is breached. Up to three escalation tiers can be configured per stage.
- Save as draft** -- The template is saved but not yet active. Review the complete stage sequence and escalation configuration.
- Publish** -- When satisfied, publish the template. Only published templates auto-assign to new contracts.

## Visual Workflow Builder

The visual workflow builder provides a drag-and-drop interface for constructing the approval stages of a template. It is the primary tool for building and editing workflow stage sequences.

### How It Works

- Adding stages** -- Click the "Add Stage" button to append a new stage to the sequence. Each stage is represented as a card in the builder.
- Stage configuration** -- Each stage card has the following fields:

Field	Description
Name	A descriptive label for the stage (e.g. "Legal Review", "Finance Sign-off").
Responsible Role	The role whose members will be assigned as approvers for this stage.
Duration (days)	The SLA for this stage -- the number of days within which the approver should act.

<b>Requires Approval</b>	A toggle that determines whether the stage requires an explicit approval action before the contract can advance to the next stage. When disabled, the stage is informational only.
--------------------------	--

- **Reordering stages** -- Drag a stage card up or down in the list to change its position in the sequence. The order in the builder is the order in which stages will be executed.
- **Removing stages** -- Click the delete button on a stage card to remove it from the sequence.
- **Preview** -- The builder shows a preview of the complete flow, allowing you to verify the stage sequence before saving or publishing.

## AI Workflow Generation

For users who prefer to describe a workflow in natural language rather than building it manually, CCRS provides an AI-powered generation feature.

### How to Use

1. On the workflow template creation or edit form, click the **"Generate with AI"** button.
2. In the text field that appears, describe the desired workflow in plain English. For example:

*"Three-stage approval: legal review for 5 days, then finance sign-off for 3 days, then executive approval for 2 days. All stages require approval."*
3. Click **Generate**. The AI processes your description and produces a stages JSON structure.
4. The generated stages appear in the visual workflow builder, where you can review, adjust, reorder, or remove them as needed.
5. Once you are satisfied with the result, save and publish the template.

### Tips for Better Results

- Be specific about stage names, responsible roles, and durations.
- Mention whether each stage requires approval or is informational.
- Include any special requirements (e.g. "the final stage should go to the CFO").
- You can regenerate as many times as needed -- the AI output replaces the current stages each time.

## Publishing and Versioning

Workflow templates follow a draft-to-published lifecycle with automatic version tracking.

### Draft State

- New templates and unpublished edits are saved in **draft** state.
- Draft templates do **not** auto-assign to new contracts.
- You can freely edit stages, escalation rules, and metadata while a template is in draft.

### Publishing

- When you publish a template, it becomes active and eligible for auto-assignment to new contracts.
- Publishing **increments the version number** automatically (v1, v2, v3, etc.).
- Previous versions are preserved in the system for audit purposes.

### Editing a Published Template

- Editing a published template creates a new draft version.
- The currently published version remains active and continues to auto-assign until the new version is published.
- Existing contract workflows that were assigned from a previous version are **not affected** by publishing a new version -- they continue using the version that was active when they were assigned.

## Template Matching

When a new contract is saved in **Draft** state, CCRS automatically searches for the most specific published workflow template that matches the contract's attributes. The matching algorithm follows a priority order from most specific to least specific:

1. **Contract type** (required) -- the template must match the contract's type (Commercial or Merchant).
2. **Project** (most specific) -- if a template is scoped to the contract's project, it takes priority.
3. **Entity** -- if no project-specific template exists, a template scoped to the contract's entity is used.
4. **Region** -- if no entity-specific template exists, a template scoped to the contract's region is used.
5. **Global** (least specific) -- if no scoped template matches, a global template (one with no region, entity, or project) is used.

If no published template matches at any level, the contract remains in **Draft** state without a workflow assignment. An administrator must create and publish a matching template before the contract can progress through its lifecycle.

## How Auto-Assignment Works

When a matching published template is found:

1. A **WorkflowInstance** is created, linking the contract to the template.
2. The contract's **workflow state** is set based on the first stage defined in the template.
3. The **approvers** for the first stage are notified per the template's role configuration.
4. The **SLA timer** begins for the first stage.

**Tip:** Create your Regions, Entities, and Projects before creating workflow templates, so you can scope them correctly. Templates cannot be scoped to organizational units that do not yet exist.

## Escalation Rules

Escalation rules ensure that contracts do not stall indefinitely at any stage. When an approver does not act within the SLA window, the system automatically escalates the item to additional stakeholders.

### Configuration

Each workflow stage can have up to **three escalation tiers**, configured independently:

Tier	Typical Threshold	Who Is Notified	Purpose
<b>Tier 1</b>	e.g. 48 hours after SLA breach	Additional stakeholders in the same role or department	Draw attention to the stalled item and encourage action.
<b>Tier 2</b>	e.g. 72 hours after SLA breach	Senior management	Involve management to resolve the delay.
<b>Tier 3</b>	e.g. 96 hours after SLA breach	Executive leadership	Final escalation for items that remain unresolved.

### How Escalation Works

1. When a stage's SLA expires without an approval or rejection action, the system triggers **Tier 1** escalation.
2. The roles specified in the Tier 1 notification list receive an alert via the configured channels (email, Microsoft Teams, or both).
3. If the item remains unresolved after the Tier 2 threshold, Tier 2 escalation fires and notifies senior management.
4. If the item still has no resolution after the Tier 3 threshold, Tier 3 escalation fires with executive-level notification.
5. Each escalation event is recorded as an **EscalationEvent** in the system for audit and reporting purposes.

### Viewing Escalated Items

- Escalated contracts appear on the **Escalations** page, accessible from the left sidebar.
- Each escalated item shows the contract name, current stage, escalation tier, time since breach, and the roles that have been notified.
- Priority indicators help you quickly identify the most critical items.

## Summary of Key Concepts

Concept	Description
Workflow Template	A reusable blueprint defining the stages, roles, SLAs, and escalation rules for contract approval.
Stage	A single step in the workflow (e.g. "Legal Review") with a responsible role, duration, and approval requirement.
Escalation Rule	A time-based trigger that notifies additional stakeholders when a stage's SLA is breached.
Workflow Instance	An active workflow linked to a specific contract, created when the template auto-assigns.
Stage Action	An approval or rejection recorded by an approver at a specific stage.
Template Matching	The process by which the system finds the most specific published template for a new contract.
Version	Each publish of a template increments its version number; previous versions are preserved.

## Role Permissions

Action	System Admin	Legal	Commercial	Finance	Operations	Audit
Create workflow template	Yes	--	--	--	--	--
Edit workflow template	Yes	--	--	--	--	--
Publish workflow template	Yes	--	--	--	--	--
Delete workflow template	Yes	--	--	--	--	--
View workflow templates	Yes	Yes	Yes	Yes	Yes	Yes
Act as stage approver	Yes	Yes	Yes	Yes	Yes	--
View escalated items	Yes	Yes	Yes	Yes	Yes	Yes

# 7. AI Analysis & Redlining

## Overview

CCRS integrates AI-powered contract analysis through a dedicated Python FastAPI microservice (the AI Worker). The AI engine can analyse contracts in five different ways, each producing structured results that are stored against the contract record. All analyses track token usage and cost in USD, giving Finance and Admin users full visibility into AI spend for budget management.

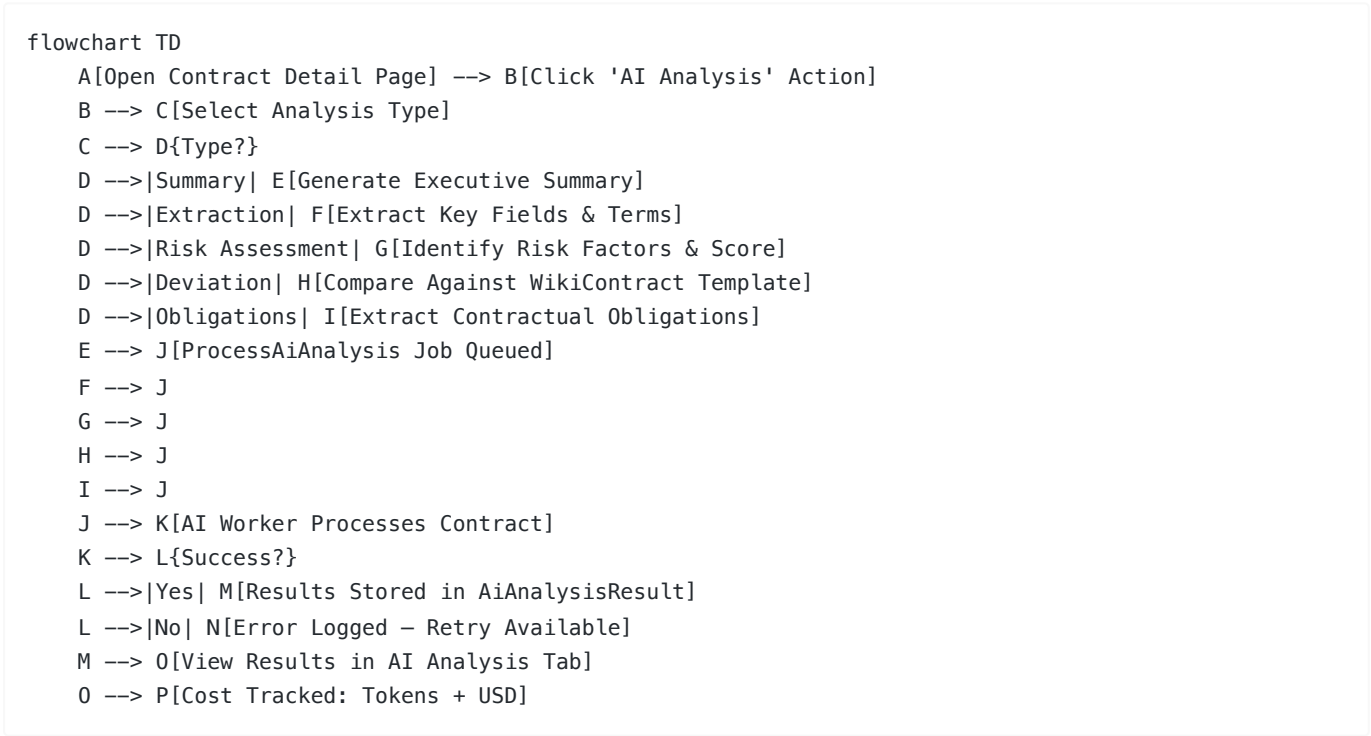
Analysis results are stored as `AiAnalysisResult` records and linked to the originating contract. For field-level extraction, individual `AiExtractedField` records capture each extracted value along with a confidence score. Every analysis records the model used, tokens consumed, and estimated cost.

## Five Analysis Types

CCRS supports five distinct AI analysis types. Each type produces a different kind of output tailored to a specific stage of the contract review process.

Analysis Type	What It Produces	Typical Use Case
Summary	An executive summary of the contract's key points, parties, terms, and obligations in plain language	Quick orientation for reviewers who need to understand a contract without reading the full document
Extraction	Structured fields (party names, dates, monetary values, terms) with a confidence score for each extracted field	Populating contract metadata, verifying that key terms match expectations
Risk Assessment	Identified risk factors, risk scores, and mitigation recommendations	Legal review to flag problematic clauses before approval
Deviation Analysis	A clause-by-clause comparison of the contract text against a WikiContract reference template, highlighting deviations from standard terms	Ensuring that negotiated contracts have not drifted from approved templates
Obligations	All contractual obligations with due dates, responsible parties, and compliance requirements	Operations and compliance teams tracking what must be delivered and by when

### AI Analysis Trigger and Results Flow



### Triggering an Analysis

Follow the steps below to run an AI analysis on a contract. Analyses can be triggered from any contract in the **Draft** or **Review** state.

#### Step-by-Step

- Navigate to the contract detail page** -- Open the contract you want to analyse from the Contracts list.
- Click the "AI Analysis" action button** -- This is available in the contract's action bar at the top of the detail page.
- Select the analysis type from the dropdown** -- Choose one of the five types: Summary, Extraction, Risk Assessment, Deviation Analysis, or Obligations.

4. **For Deviation Analysis, select the WikiContract template** -- A dropdown lists all available WikiContract reference templates. Choose the template that represents the standard terms you want to compare against.
5. **Click "Run Analysis"** -- The analysis is submitted.
6. **Monitor the status** -- The analysis is queued as a background job. You will see the status change through three stages:
- **Pending** -- The job has been created and is waiting in the queue.
  - **Processing** -- The AI Worker microservice is actively analysing the contract.
  - **Completed** -- Results are ready to view.
  - If the analysis fails, the status changes to **Failed** with an error message. You can retry the analysis from the same screen.
7. **View results** -- Once completed, results appear in the **AI Analysis** tab on the contract detail page.

You can run multiple analysis types on the same contract. Each analysis creates a separate result record.

---

## Viewing Results

Analysis results are displayed in the **AI Analysis** relation manager tab on the contract detail page. This tab lists all analyses that have been run on the contract.

### Results List

Each result row shows:

Column	Description
Analysis Type	Summary, Extraction, Risk Assessment, Deviation, or Obligations
Status	Pending, Processing, Completed, or Failed
Date	When the analysis was triggered
Model Used	The AI model that processed the analysis
Tokens Used	The number of tokens consumed by the analysis
Cost (USD)	The estimated cost of the analysis in US dollars

### Viewing Full Results

Click on any completed result to view the full analysis output. The output format varies by analysis type:

- **Summary** -- A structured narrative covering key parties, terms, obligations, and notable provisions.
- **Extraction** -- A table of extracted fields. Each field shows the field name, extracted value, and a confidence score (0-100%). Higher confidence scores indicate greater certainty in the extraction.
- **Risk Assessment** -- A list of identified risk factors, each with a severity score and recommended mitigation actions.
- **Deviation Analysis** -- A comparison showing which clauses deviate from the WikiContract template, with the original template text alongside the contract text.
- **Obligations** -- A structured list of obligations with due dates, responsible parties, and compliance criteria.

### Failed Analyses

If an analysis fails, the result record includes an **error message** explaining the failure. Common causes include:

- The contract document could not be parsed (corrupted or unsupported format).
- The AI Worker microservice was temporarily unavailable.
- The contract text exceeded the model's context window.

Click **Retry** on a failed analysis to re-queue the job.

---

## Cost Tracking

Every AI analysis records the number of tokens consumed and the estimated cost in USD. This data is available at the individual result level and in aggregate through the AI Cost Report.

### Per-Analysis Cost

Each `AiAnalysisResult` record stores:

- **tokens\_used** -- The total token count (input + output) for the analysis.
- **cost\_usd** -- The estimated cost based on the model's per-token pricing.
- **model\_used** -- The specific AI model that processed the request.

### AI Cost Report Page

The **AI Cost Report** page is available under the Reports navigation group. Access is restricted to users with the **Finance** or **System Admin** role.

The report provides:

- **Breakdown by analysis type** -- See which analysis types consume the most tokens and cost.
- **Breakdown by contract** -- Identify which contracts have required the most AI processing.
- **Breakdown by time period** -- Track spending trends over days, weeks, or months.
- **Total spend** -- Aggregate cost across all analyses for budget monitoring.

Use this report to manage AI usage across the organisation and set expectations for monthly AI processing budgets.

---

## Redline Review

Redline Review is an AI-powered clause-by-clause comparison of a contract against a reference WikiContract template. It produces actionable recommendations for each clause, helping Legal teams quickly identify non-standard terms and decide how to proceed.

### When to Use Redline Review

Use Redline Review when:

- You are reviewing a counterparty-drafted contract against your standard terms.
- You need to identify which clauses deviate from an approved template.
- Legal wants a structured, clause-level comparison rather than a high-level deviation summary.

Redline Review is more granular than Deviation Analysis. While Deviation Analysis provides an overview of differences, Redline Review breaks the comparison down to individual clauses with specific recommendations.

### Redline Review Session Flow

flowchart TD

```
A[Open Contract Detail Page] --> B[Click 'Start Redline Review']
B --> C[Select WikiContract Template as Reference]
C --> D[RedlineService Creates Session]
D --> E[AI Performs Clause-by-Clause Comparison]
E --> F[Each Clause Analyzed]
F --> G{AI Recommendation?}
G -->|Accept| H[Clause Matches Template – No Action Needed]
G -->|Modify| I[Suggested Changes Provided]
G -->|Reject| J[Clause Flagged for Removal/Rewrite]
H --> K[Results Displayed on Redline Session Page]
I --> K
J --> K
```

```
J --> K
K --> L[User Reviews Each Clause]
L --> M[Accept or Override AI Recommendation]
M --> N[Session Complete – Summary Generated]
```

## How to Run a Redline Review

1. **Open the contract detail page** -- Navigate to the contract you want to redline.
2. **Click "Start Redline Review"** -- This action is available for contracts in Draft or Review state.
3. **Select the WikiContract template** -- Choose the reference template that represents your standard terms. The template serves as the baseline for comparison.
4. **Wait for processing** -- The RedlineService creates a session and sends each clause to the AI for comparison. Processing time depends on the length of the contract.
5. **Review the results** -- Once complete, the Redline Session page displays each clause with a side-by-side comparison.

## Understanding Redline Results

Each clause in the session receives one of three AI recommendations:

Recommendation	Meaning	Action Required
<b>Accept</b>	The clause matches or is substantially equivalent to the template clause.	No action needed -- the clause conforms to standard terms.
<b>Modify</b>	The clause differs from the template in ways that may need adjustment. The AI provides suggested changes.	Legal reviews the suggested modification and decides whether to incorporate it.
<b>Reject</b>	The clause is significantly different from standard terms and may need to be removed or completely rewritten.	Legal evaluates whether the clause is acceptable or must be renegotiated.

## Reviewing and Overriding Recommendations

For each clause in the session:

- Review the **original text** (from the WikiContract template) alongside the **contract text**.
- Read the AI's **analysis** explaining why it made the recommendation.
- **Accept the recommendation** to mark the clause as reviewed and agree with the AI's assessment.
- **Override the recommendation** if you disagree with the AI's assessment. You can change the status of any clause regardless of what the AI recommended.

Once all clauses have been reviewed, the session is marked as complete and a summary of findings is generated.

## WikiContract Templates

WikiContract templates are reference documents that represent your organisation's standard contract terms. They serve as baselines for both Deviation Analysis and Redline Review.

### Purpose

- Establish a single source of truth for standard contractual language.
- Enable consistent comparison across all contracts reviewed by the organisation.
- Reduce legal review time by automating the identification of non-standard clauses.

### Management

WikiContract templates are managed by users with the **Legal** or **System Admin** role. Key capabilities include:

- **Create templates** -- Draft new WikiContract templates covering different contract types or jurisdictions.

- **Edit templates** -- Update template language as standard terms evolve.
- **Template signing blocks** -- Define field positioning for signature blocks, specifying where signing fields should appear during the electronic signing process.
- **Version control** -- Maintain current versions of templates so that analyses always compare against the latest approved standard terms.

Using Templates

WikiContract templates are referenced in two contexts:

1. **Deviation Analysis** -- When running a Deviation Analysis on a contract, select the appropriate WikiContract template to compare against. The AI highlights where the contract's terms diverge from the template.
2. **Redline Review** -- When starting a Redline Review session, select the WikiContract template as the baseline. The AI performs a clause-by-clause comparison and provides Accept, Modify, or Reject recommendations for each clause.

Choose the template that most closely matches the contract type being reviewed. For example, use a merchant agreement template when reviewing merchant contracts and a vendor services template when reviewing vendor agreements.

Permissions

The table below summarises which roles can access AI analysis and redlining features.

Capability	System Admin	Legal	Commercial	Finance	Operations	Audit
Trigger AI analysis	Yes	Yes	No	No	No	No
View AI analysis results	Yes	Yes	Yes	View only	View only	View only
Start Redline Review	Yes	Yes	No	No	No	No
Review redline clauses	Yes	Yes	No	No	No	No
View AI Cost Report	Yes	No	No	Yes	No	No
Manage WikiContract templates	Yes	Yes	No	No	No	No

Tips and Best Practices

- **Run Summary first** -- Before diving into detailed analysis, run a Summary to get a quick understanding of the contract's key terms and parties.
- **Use Extraction to populate metadata** -- After uploading a contract, run Extraction to automatically identify party names, dates, and values. Verify the extracted fields against the document.
- **Check confidence scores** -- Extraction results include confidence scores for each field. Review any field with a confidence score below 80% manually.
- **Choose the right template** -- For Deviation Analysis and Redline Review, selecting the correct WikiContract template is critical. An incorrect template will produce misleading comparisons.
- **Monitor costs** -- Review the AI Cost Report regularly. If costs are higher than expected, check whether analyses are being run redundantly on the same contracts.
- **Retry failed analyses** -- Transient failures (such as the AI Worker being temporarily unavailable) can usually be resolved by retrying the analysis.

8. Reports & Analytics

Overview

CCRS provides three dedicated reporting surfaces -- the **Reports Page** for tabular contract data with export capabilities, the **Analytics Dashboard** for visual insights via interactive widgets, and the **AI Cost Report** for monitoring AI analysis spending. In addition, every user sees summary widgets on the **Main Dashboard** upon login.

## Reports Page

**Access:** Finance, Legal, Audit, System Admin

The Reports Page presents a filterable, sortable table of all contracts in the system. Use it to locate specific contracts, review portfolio composition, and generate exports for external stakeholders.

### Navigating to Reports

1. Open the **Reports** navigation group in the left sidebar.
2. Click **Reports**.

### Table Columns

The reports table displays the following columns for each contract:

Column	Description
Title	The contract name (searchable)
Type	Contract type displayed as a badge (e.g., Commercial, Merchant)
Counterparty	The legal name of the counterparty (searchable)
Region	The geographic region the contract belongs to
Entity	The legal entity entering into the agreement
State	Current workflow state shown as a colour-coded badge
Expiry	Contract expiry date (sortable)
Created	Date the contract was created (sortable)

Click any sortable column header to reorder results. The table defaults to most recently created contracts first.

### Filtering

Filters appear above the table and update results in real time as you change them. You do not need to click a "search" button -- the table refreshes automatically.

Filter	Options
State	Draft, Review, Approval, Signing, Countersign, Executed, Archived
Type	Dynamically populated from contract types in the database
Region	All configured regions
Entity	All configured entities

Multiple filters can be combined. For example, selecting State = "Executed" and Region = "EMEA" shows only executed contracts in the EMEA region.

To clear all filters, click the reset icon next to the filter bar.

### Pagination

The table supports configurable page sizes of 10, 25, 50, or 100 rows. Use the dropdown at the bottom of the table to adjust how many contracts are displayed per page.

## Exporting Data

Two export options are available from the Reports Page header. Both export buttons respect whatever filters are currently applied -- only the contracts visible in the filtered results are included in the export file.

### Report Export Flow

```
graph LR
    A[Navigate to Reports Page] --> B[Apply Filters]
    B --> C{Filter Options}
    C --> D[By State: Draft/Review/Approval/etc.]
    C --> E[By Type: Commercial/Merchant]
    C --> F[By Region/Entity/Project]
    C --> G[By Date Range]
    D --> H[Filtered Results Displayed]
    E --> H
    F --> H
    G --> H
    H --> I{Export Format?}
    I -->|Excel| J[Generate .xlsx File]
    I -->|PDF| K[Generate PDF Report]
    J --> L[Browser Downloads File]
    K --> L
```

### Export to Excel

Click the green **Export Excel** button in the page header. CCRS generates a comprehensive `.xlsx` spreadsheet containing all filtered contracts with their full details. The file opens in a new browser tab and downloads automatically.

#### Use cases:

- Importing contract data into financial models or ERP systems.
- Building pivot tables for ad-hoc analysis.
- Sharing raw data with colleagues who need to perform their own calculations.

### Export to PDF

Click the red **Export PDF** button in the page header. CCRS generates a formatted PDF report suitable for printing or attaching to emails and board packs. The file opens in a new browser tab and downloads automatically.

#### Use cases:

- Producing a snapshot report for management review.
- Attaching a portfolio summary to compliance submissions.
- Printing a hard copy for physical filing or meeting handouts.

## Analytics Dashboard

**Access:** System Admin, Legal, Finance, Audit (requires the **advanced\_analytics** feature flag to be enabled)

The Analytics Dashboard is a visual reporting surface designed for executive-level insight into the contract portfolio. It is accessed via the **Analytics Dashboard** item in the Reports navigation group.

## Widgets

The dashboard displays six interactive widgets arranged in a two-column layout:

### Contract Pipeline Funnel

A visual funnel chart showing how many contracts are at each lifecycle stage (Draft, Review, Approval, Signing, Countersign, Executed, Archived). The funnel shape makes it easy to spot bottlenecks -- if contracts are accumulating at the Approval stage, for example, additional approvers may be needed.

### Risk Distribution

A breakdown of contracts by AI-assessed risk level: **Low**, **Medium**, **High**, and **Critical**. This widget provides a quick health check on the overall portfolio risk profile. Contracts are categorised based on results from the AI risk analysis feature.

### Compliance Overview

A summary of regulatory compliance status across the contract portfolio. Shows the proportion of contracts that are compliant, non-compliant, or pending compliance review against mapped regulatory frameworks. This widget is especially relevant for Legal and Audit teams monitoring regulatory exposure.

### Obligation Tracker

Displays upcoming contractual obligations and deadlines with priority indicators. Obligations are extracted from contracts (either manually entered or identified by AI analysis) and sorted by urgency. This helps ensure that deliverables, payment milestones, and renewal decisions are actioned on time.

### AI Usage & Cost

A spending overview for AI-powered contract analysis. Shows total cost, number of analyses, and cost trends over time. Useful for Finance teams monitoring AI feature budgets.

### Workflow Performance

Metrics on average time spent at each workflow stage, helping identify where the approval process slows down. If contracts routinely spend weeks in the Approval stage but only days in Review, this widget highlights the imbalance so process owners can investigate.

---

## Main Dashboard Widgets

**Access:** All roles (visible on the home dashboard after login)

The main Dashboard is the first screen every user sees. It provides a summary of the most important metrics through six core widgets (with a seventh appearing when the regulatory compliance feature is enabled).

### Contract Status

A chart showing the distribution of contracts across workflow states. At a glance, you can see how many contracts are in Draft, Review, Approval, Signing, Countersign, Executed, and Archived states.

### Expiry Horizon

Lists contracts expiring within the next **30**, **60**, and **90 days**. This widget ensures that upcoming expirations are visible well in advance, giving Commercial and Operations teams time to initiate renewals or renegotiations.

### Pending Workflows

Shows your personal action items -- contracts that are waiting for your review, approval, or signature. Each item links directly to the relevant contract so you can take action immediately.

### Active Escalations

Highlights overdue workflow stages that need management attention. When a contract has been sitting in a workflow stage past its expected completion time, it appears here as an escalation.

AI Cost

A summary card showing total AI analysis spending -- the number of analyses run and the cumulative cost in USD. This gives a quick pulse on AI feature usage without navigating to the full AI Cost Report.

Obligation Tracker

Surfaces upcoming obligations and deadlines from across the contract portfolio. Functions identically to the Analytics Dashboard version but is included on the main Dashboard for broader visibility.

Compliance Overview (conditional)

Appears only when the **regulatory\_compliance** feature flag is enabled. Shows the same compliance status summary as the Analytics Dashboard widget.

AI Cost Report

**Access:** Finance, System Admin

The AI Cost Report provides a detailed, line-item breakdown of every AI analysis performed in CCRS. It is accessed via the **AI Cost Analytics** item in the Reports navigation group.

What It Tracks

Each row in the report represents a single AI analysis run and includes the following data:

Column	Description
Contract	The contract that was analysed (searchable)
Analysis Type	The type of analysis performed (Summary, Extraction, Risk, Template Deviation, Obligations)
Model	The AI model used for the analysis
Input Tokens	Number of tokens sent to the AI model
Output Tokens	Number of tokens received from the AI model
Cost (USD)	Dollar cost of the analysis
Time (ms)	Processing time in milliseconds
Status	Completion status (Completed, Failed, Processing, Pending)
When	Relative timestamp (e.g., "2 hours ago")

Filters

Filter	Options
Analysis Type	Summary, Extraction, Risk Analysis, Template Deviation, Obligations
Status	Completed, Failed, Processing, Pending

Summary Statistics

The page header displays four aggregate metrics calculated from all completed analyses:

- **Total Cost** -- cumulative USD spend across all analyses.
- **Total Tokens** -- combined input and output tokens consumed.
- **Total Analyses** -- number of completed analysis runs.
- **Average Cost** -- mean USD cost per analysis.

These figures help Finance teams budget for AI features and identify whether usage is trending up or down.

## Who Can Access What

The table below summarises reporting and analytics access by role.

Report / Feature	System Admin	Legal	Commercial	Finance	Operations	Audit
Reports Page	Yes	Yes	--	Yes	--	Yes
Export Excel / PDF	Yes	Yes	--	Yes	--	Yes
Analytics Dashboard	Yes	Yes	--	Yes	--	Yes
AI Cost Report	Yes	--	--	Yes	--	--
Main Dashboard	Yes	Yes	Yes	Yes	Yes	Yes

**Notes:**

- The Analytics Dashboard requires the **advanced\_analytics** feature flag to be enabled. When disabled, the navigation item is hidden and the page is inaccessible regardless of role.
- Export buttons on the Reports Page are available to every role that can access the Reports Page.
- The Main Dashboard is visible to all authenticated users and adapts its content based on the user's role and permissions.

## Tips

- **Narrow before exporting** -- Apply filters to limit results to the contracts relevant to your report. Exporting the entire portfolio into Excel produces a large file; filtering first keeps exports focused and manageable.
- **Bookmark filtered views** -- Filters are encoded in the URL. Bookmark a filtered Reports Page to return to the same view without re-applying filters.
- **Use the Analytics Dashboard for presentations** -- The visual widgets are designed for executive summaries. Screenshot them or share the page URL with stakeholders who have the appropriate role.
- **Monitor AI costs regularly** -- If your organisation runs AI analysis on every contract, costs can accumulate. Use the AI Cost Report to establish a baseline and track month-over-month trends.

# 9. Bulk Operations

## Overview

System Admins can bulk-import data and contracts via CSV files. This is useful for initial system setup, migrating data from other systems, or batch-processing large numbers of records.

Both bulk operations are restricted to users with the **system\_admin** role. They are found under the **Administration** section of the main navigation.

## Bulk Data Upload

**Location:** Administration > Bulk Data Upload

The Bulk Data Upload page lets you import reference data into CCRS in batches. Instead of creating records one at a time through the UI, you can prepare a CSV file and upload hundreds of records at once.

Supported Data Types

Data Type	Key Columns	Notes
Regions	name, code	Geographic groupings. Must be created before Entities.
Entities	name, code, region_code	Legal entities within a Region. Must be created before Projects.
Projects	name, code, entity_code	Business projects within an Entity.
Users	name, email, role	User accounts provisioned in the system.
Counterparties	legal_name, registration_number, jurisdiction	External parties to contracts.

Step-by-Step Instructions

1. **Select the data type** from the dropdown at the top of the page.
2. **Click "Download Template"** to get a CSV file with the correct column headers for that data type.
3. **Fill in the template** with your data. Each row represents one record. Do not modify the header row.
4. **Upload the completed CSV file** using the file upload field.
5. The system validates each row and processes all valid ones.
6. **Review the results:** the page displays the success count, the failure count, and detailed error messages for any rows that failed.

Import Order and Dependencies

You must create parent records before child records that reference them:

1. **Regions first** -- these have no dependencies.
2. **Entities second** -- each Entity references a Region via the `region_code` column.
3. **Projects third** -- each Project references an Entity via the `entity_code` column.

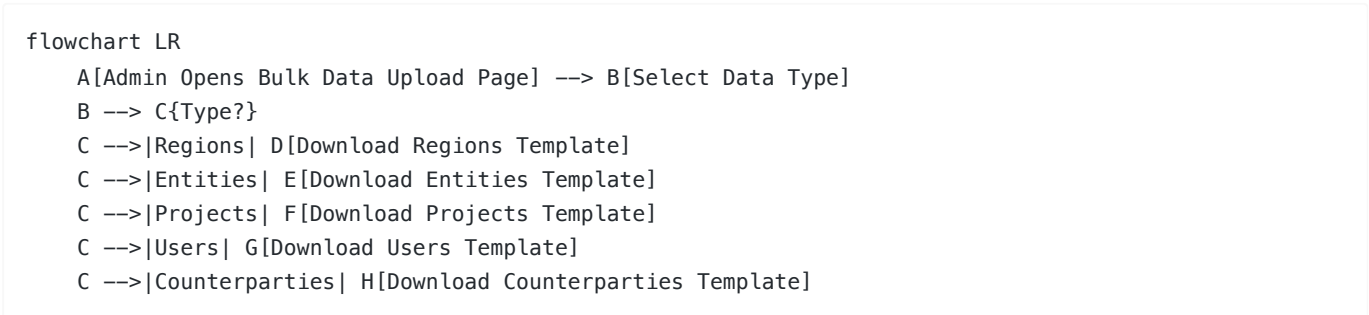
Users and Counterparties have no ordering dependencies and can be imported at any time.

Important Notes

- The `region_code` and `entity_code` columns expect the **code** value (e.g., `EMEA` , `DTL-UK` ), not the name and not the database ID.
- Duplicate checking is performed automatically. If a record with the same code already exists, that row is skipped rather than creating a duplicate.
- The CSV file must be UTF-8 encoded. If you prepare it in Excel, use "Save As > CSV UTF-8" to avoid character encoding issues.

Bulk Data Import Flow

The following diagram shows the end-to-end process for importing reference data:



```
D --> I[Fill in CSV Template]
E --> I
F --> I
G --> I
H --> I
I --> J[Upload Completed CSV]
J --> K[BulkDataImportService Validates Rows]
K --> L{Validation Results}
L --> M[Success Count: N records created]
L --> N[Failure Count: M rows with errors]
N --> O[Error Report: Row number + Error message]
```

## Bulk Contract Upload

**Location:** Administration > Bulk Contract Upload

The Bulk Contract Upload page is designed for importing many contracts at once, each with an associated document file. You provide a CSV manifest that describes the contracts and a ZIP archive containing the actual files.

### Preparation

Before uploading, you need to prepare two files:

#### 1. CSV Manifest

Create a CSV file with the following columns:

Column	Description	Example
title	Contract title	"Service Agreement - Acme Corp"
contract_type	Type of contract	"service_agreement"
region_code	Code of the Region	"EMEA"
entity_code	Code of the Entity	"DTL-UK"
project_code	Code of the Project	"PROJ-001"
counterparty_registration	Registration number of the counterparty	"12345678"
filename	Name of the file in the ZIP archive	"acme-service-agreement.pdf"
start_date	Contract start date (YYYY-MM-DD)	"2026-03-01"
end_date	Contract end date (YYYY-MM-DD)	"2027-02-28"
value	Contract monetary value	"50000.00"
currency	Currency code	"GBP"

#### 2. ZIP Archive

Create a ZIP file containing all the contract documents (PDF or DOCX). The filename of each document must match exactly what appears in the `filename` column of the CSV manifest. Filenames are case-sensitive.

### Upload Limits

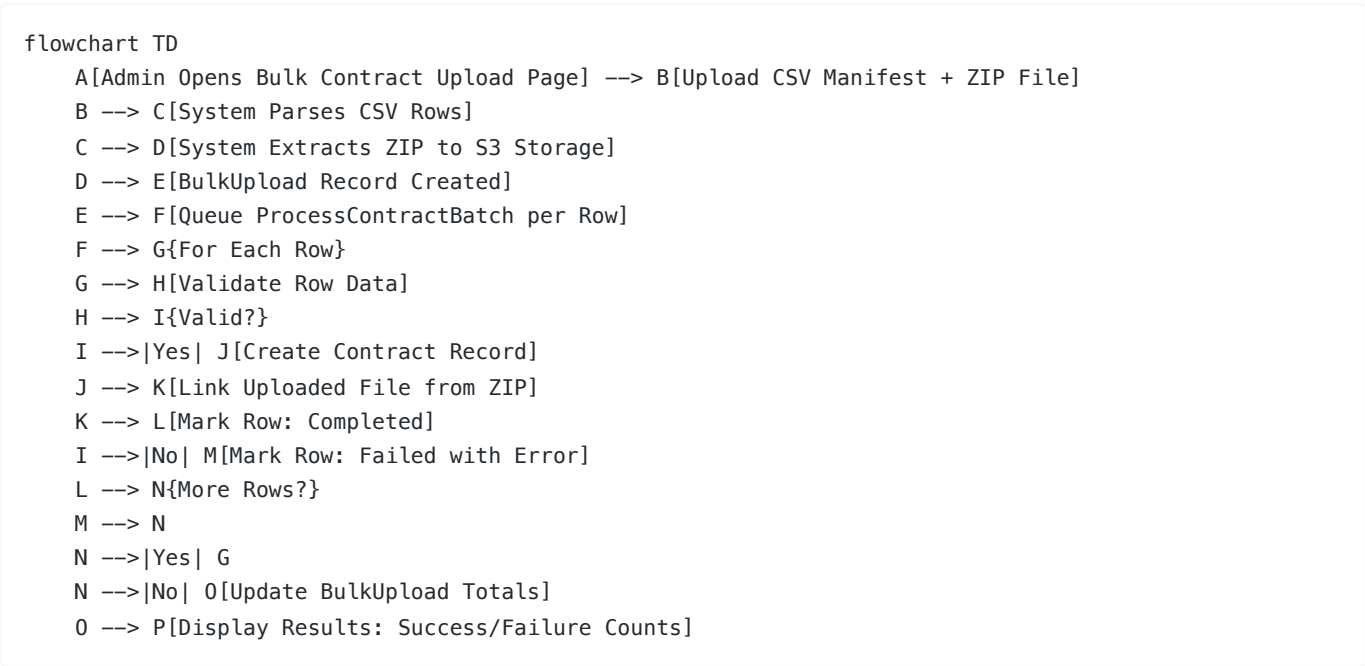
- **Maximum 500 files** per upload.
- **Maximum 50 MB** per individual file within the ZIP.

Upload Process

- 1. Navigate to the **Bulk Contract Upload** page under Administration.
- 2. Upload the **CSV manifest** file.
- 3. Upload the **ZIP file** containing the contract documents.
- 4. Click **"Process Upload"** to begin.
- 5. The system validates the manifest, extracts files from the ZIP, and creates contract records.
- 6. A progress display shows real-time status as each row moves from pending to processing to completed or failed.
- 7. Review the results: each row shows its individual status and any error messages.

Bulk Contract Upload Processing Flow

The following diagram shows how the system processes a bulk contract upload:



Monitoring Progress

Bulk uploads are processed **asynchronously** via background jobs. You do not need to keep the page open for processing to continue, but the page provides real-time feedback if you stay on it.

Status Tracking

Each bulk upload has an overall status:

Status	Meaning
Pending	Upload received, waiting for processing to begin.
Processing	Background jobs are actively working through the rows.
Completed	All rows have been processed (some may have failed individually).
Failed	The upload itself failed (e.g., invalid CSV format, corrupt ZIP).

Each individual row within the upload is also tracked with its own status (pending, processing, completed, or failed).

Summary Totals

The upload detail view shows:

- **Total Rows** -- the number of data rows in the CSV.
- **Successful** -- the number of rows that were processed and created records.
- **Failed** -- the number of rows that encountered errors.

## Error Logs

For failed rows, the system records a detailed error log in JSON format. This log includes the row number, the data that was submitted, and the specific validation or processing error that occurred. You can review this information directly on the upload detail page.

## Troubleshooting Common Errors

Error	Cause	Fix
"Region code not found"	The <code>region_code</code> value in the CSV does not match any existing Region.	Verify region codes against the Regions list. Import Regions first if needed.
"Entity code not found"	The <code>entity_code</code> value in the CSV does not match any existing Entity.	Verify entity codes against the Entities list. Import Entities first if needed.
"File not found in ZIP"	The <code>filename</code> in the CSV does not match any file in the ZIP archive.	Check that filenames match exactly, including case and file extension.
"Duplicate record"	A record with the same code or unique identifier already exists.	Remove the duplicate row from the CSV, or update the existing record manually.
"Invalid file format"	A file in the ZIP is not a PDF or DOCX.	Replace the file with a PDF or DOCX version. Only these formats are accepted.
"File exceeds size limit"	An individual file in the ZIP is larger than 50 MB.	Compress or split the file to bring it under the 50 MB limit.
"Too many files"	The CSV manifest contains more than 500 rows.	Split the upload into multiple batches of 500 or fewer.

## Best Practices

- **Start with a small test batch.** Upload 5-10 records first to confirm your CSV format is correct before processing a large file.
- **Follow the dependency order.** Always import Regions before Entities, and Entities before Projects. Contracts depend on all three plus Counterparties.
- **Use the downloaded templates.** The templates include the exact column headers the system expects. Adding, removing, or renaming columns will cause validation failures.
- **Check codes, not names.** The CSV columns `region_code`, `entity_code`, and `project_code` require the short code value, not the human-readable name.
- **Verify ZIP contents.** Before uploading, open the ZIP and confirm that every filename listed in your CSV manifest is present and spelled correctly.
- **Review error reports promptly.** After a bulk upload completes, check the error log for any failed rows. Fix and re-upload only the failed rows in a new CSV rather than re-uploading the entire batch.

# 10. Notifications & Reminders

CCRS keeps you informed about contract activity through a multi-channel notification system. You can receive alerts via email, Microsoft Teams, the in-app notifications panel, or as downloadable calendar events -- and you control exactly which channels each notification category uses.

## Notification Channels

CCRS supports four notification channels. Each serves a different workflow need.

Channel	How It Works	Best For
Email	Standard email notifications sent to the address registered on your account.	Formal alerts you need a paper trail for -- approvals, escalations, signing events.
Microsoft Teams	Posts to a configured Teams channel via webhook integration.	Keeping your team informed of contract activity in real time without leaving Teams.
In-App	Notifications appear in the <b>bell icon</b> in the top-right corner of every CCRS page. A badge shows the count of unread items.	Quick, low-friction alerts while you are already working inside CCRS.
Calendar (ICS)	Generates a downloadable .ics calendar file for date-based notifications. The file can be imported into Outlook, Google Calendar, Apple Calendar, or any standards-compliant calendar application.	Key date reminders -- contract expiry, renewal deadlines, payment milestones.

## Notification Preferences

You can control exactly which types of notifications you receive and on which channels.

### Accessing Preferences

1. Click your **name or avatar** in the top-right corner of the page.
2. Select **Notification Preferences** from the profile menu.

### Configuring Preferences

The Notification Preferences page displays a grid of **notification categories** along the left and **channels** across the top. Toggle each combination on or off to match your workflow.

Available categories include:

- **Workflow Actions** -- stage assignments, approval requests, rejections.
- **Contract Updates** -- status changes, field edits, document uploads.
- **Signing Events** -- signing session creation, signature completion, countersigning requests.
- **Escalations** -- SLA breaches and tiered escalation alerts.
- **Reminders** -- key date reminders (expiry, renewal, payment, custom dates).

For example, you might enable email and Teams for **Escalations** (so you never miss a time-critical alert) but limit **Contract Updates** to in-app only (to avoid inbox clutter).

## The Notifications Inbox

The in-app notifications inbox is your central hub for all CCRS notifications regardless of which other channels are enabled.

### Viewing Notifications

1. Click the **bell icon** in the top-right corner of any page.
2. A dropdown panel shows your most recent notifications, newest first.
3. Click a notification to navigate directly to the relevant record (contract, workflow stage, signing session, etc.).

### Marking as Read

- Clicking a notification marks it as **read** automatically.
- You can also mark individual notifications or **all notifications** as read from the panel.

## Notification Data

Each notification contains:

Field	Description
<b>Type</b>	The category of the notification (workflow, signing, escalation, etc.).
<b>Title</b>	A brief summary of the event.
<b>Message</b>	A more detailed description of what happened and what action (if any) is required.
<b>Channel</b>	Which channel delivered this notification (email, Teams, in-app, calendar).
<b>Timestamp</b>	When the notification was created.
<b>Read Status</b>	Whether you have viewed the notification.

---

## Key Dates

Key Dates are specific milestones associated with a contract -- expiry dates, renewal windows, payment due dates, compliance review dates, and any custom dates your organisation tracks.

### Viewing Key Dates

1. Navigate to **Key Dates** in the left sidebar.
2. The Key Dates page shows a consolidated list of upcoming milestones across **all contracts** you have access to.

### Key Date Fields

Field	Description
<b>Contract</b>	The contract this date belongs to. Click to open the contract record.
<b>Date Type</b>	The category of the date -- e.g. Expiry, Renewal, Payment, Custom.
<b>Date Value</b>	The actual date.
<b>Description</b>	Additional context about the milestone.

### Filtering

Use the filters at the top of the page to narrow the list by:

- **Date range** -- show only dates within a specific window.
- **Contract** -- focus on a single contract's milestones.
- **Date type** -- show only expiry dates, only payment dates, etc.

Key dates link directly to their associated contract, so you can navigate to the full contract record in a single click.

---

## Reminders

Reminders are the mechanism that turns Key Dates into proactive notifications. Instead of checking the Key Dates page manually, you configure reminders to alert you automatically as a date approaches.

### How Reminders Work

Each reminder is linked to a specific **Contract Key Date** and defines:

- **Lead Days** -- how many days before the key date the reminder should fire. For example, a lead of 30 days on a contract expiry date means you are notified one month before the contract expires.
- **Channel** -- which notification channel the reminder uses (email, Teams, in-app, or calendar).
- **Active / Inactive** -- a toggle to enable or disable the reminder without deleting it.

When the current date falls within the lead-day window, CCRS automatically dispatches the reminder through the configured channel and records the send timestamp.

Creating a Reminder

1. Navigate to **Reminders** in the left sidebar.
2. Click **"New Reminder"**.
3. Select the **Contract** and **Key Date** the reminder applies to.
4. Set the **Lead Days** value (e.g. 7, 14, 30, 60, 90).
5. Choose the **Channel** for delivery.
6. Ensure the **Active** toggle is on.
7. Click **Save**.

Managing Reminders

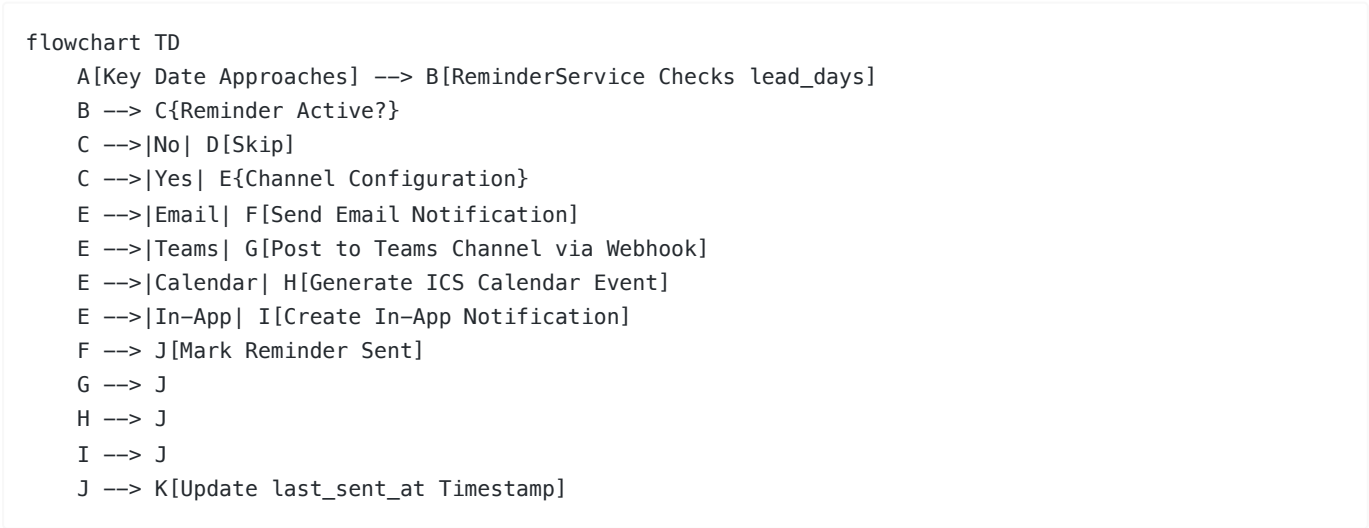
The Reminders page lists all configured reminders with their current status.

Column	Description
Contract	The associated contract.
Key Date	The milestone this reminder tracks.
Lead Days	Days before the date when the reminder fires.
Channel	Delivery channel (email, Teams, in-app, calendar).
Active	Whether the reminder is currently enabled.
Last Sent At	Timestamp of the most recent dispatch -- useful for confirming the reminder is working.

You can edit, deactivate, or delete reminders at any time.

Reminder and Notification Dispatch Flow

The diagram below shows how CCRS processes reminders and dispatches notifications as key dates approach.



## Flow Explanation

1. **Key Date Approaches** -- the system periodically checks all active reminders against the current date.
  2. **Lead Days Check** -- for each reminder, CCRS calculates whether the current date falls within the lead-day window before the key date.
  3. **Active Check** -- inactive reminders are skipped entirely.
  4. **Channel Dispatch** -- the notification is sent through the channel configured on the reminder (email, Teams, calendar ICS, or in-app).
  5. **Mark Sent** -- the reminder's `last_sent_at` timestamp is updated to prevent duplicate sends.
- 

## Escalation Notifications

Escalation notifications are a special category that is triggered automatically by the workflow engine -- you do not configure these manually.

### When Escalations Occur

An escalation is generated when a workflow stage's **SLA is breached** -- that is, the assigned approver has not taken action within the allotted time. Escalations follow the **tiered system** defined in the workflow template (see [Chapter 6 -- Workflow Templates](#) for details on configuring tiers).

### How Escalation Notifications Are Delivered

- Escalation notifications follow the same multi-channel system as all other notifications.
- They are delivered to the channels you have enabled for the **Escalations** category in your Notification Preferences.
- Escalation events also appear on the **Escalations** page (accessible from the left sidebar), where they are displayed with **priority indicators** so you can triage by urgency.

### Escalation Priority Levels

Priority	Meaning
<b>Tier 1</b>	Initial escalation -- the original approver has missed the SLA. Additional stakeholders are notified.
<b>Tier 2</b>	Continued inaction -- senior management is brought into the loop.
<b>Tier 3</b>	Executive escalation -- the highest escalation tier, typically involving department heads or executive leadership.

Each tier adds progressively more senior recipients to ensure that no contract action is left unattended indefinitely.

---

## Microsoft Teams Integration

CCRS integrates with Microsoft Teams via **incoming webhooks**. When Teams is enabled as a notification channel, CCRS posts formatted messages to the configured Teams channel.

### What Teams Notifications Look Like

Teams notifications include:

- A **title** summarising the event.
- A **message body** with relevant details (contract title, reference number, key date, etc.).
- A **link** back to the relevant record in CCRS for one-click navigation.

### Configuration

Teams webhook URLs are configured by your **System Administrator** at the organisation level. If you need Teams notifications but are not receiving them, verify that:

1. Teams is enabled in your **Notification Preferences** for the relevant categories.

2. The Teams webhook has been configured by your System Administrator.

---

## Calendar (ICS) Integration

For date-driven notifications -- particularly key date reminders -- CCRS can generate **ICS calendar files** that you download and import into your preferred calendar application.

### Supported Calendar Applications

Any application that supports the iCalendar (.ics) standard, including:

- Microsoft Outlook
- Google Calendar
- Apple Calendar
- Thunderbird

### How It Works

When a reminder or notification is dispatched via the **Calendar** channel:

1. CCRS generates an `.ics` file containing the event details (date, title, description, contract reference).
2. The file is made available for download.
3. Import the file into your calendar application to create a calendar event with the appropriate date and details.

This is especially useful for tracking contract expiry dates, renewal windows, and payment milestones alongside your regular schedule.

---

## Best Practices

- **Enable at least two channels for critical categories.** For escalations and signing events, consider enabling both email and Teams (or email and in-app) so that you have a backup if one channel is missed.
  - **Use calendar ICS for key dates.** Importing expiry and renewal dates into your calendar ensures they appear alongside your meetings and other commitments.
  - **Set multiple reminders with staggered lead days.** For high-value contracts, create reminders at 90, 60, 30, and 7 days before expiry to give yourself escalating urgency.
  - **Review your preferences periodically.** As your role or responsibilities change, revisit Notification Preferences to ensure you are still receiving the right alerts on the right channels.
  - **Check the Notifications inbox regularly.** Even if you rely primarily on email or Teams, the in-app inbox serves as a single source of truth for all notifications.
- 

# 11. Organization Setup

CCRS organizes contracts using a three-level hierarchy: **Regions**, **Entities**, and **Projects**. This structure determines how workflow templates are matched to contracts, how reports can be filtered, and how access is scoped across the organisation. Only **System Admin** users can create, edit, or delete organisation structure records.

---

## Organizational Hierarchy

The hierarchy flows from broad geographic groupings down to specific operational units.

```
Region (e.g. MENA, EMEA, APAC)
├── Entity (e.g. Digittal AE, Digittal UK)
│   └── Project (e.g. PRJ-001, PRJ-002)
```

Every contract in CCRS is associated with a **Project**, which belongs to an **Entity**, which belongs to a **Region**. This chain gives you consistent, multi-level filtering and reporting throughout the system.

## Why This Matters

- **Workflow template matching** -- templates can be scoped to a specific Region, Entity, or Project. When a new contract is created, CCRS selects the most specific matching template.
- **Report filtering** -- dashboards and reports can be filtered by any combination of Region, Entity, and Project.
- **Access scoping** -- signing authorities and permissions can be restricted to specific parts of the hierarchy.

---

## Setup Order

You must create records in dependency order. Each level depends on the one above it.

Step	Record Type	Depends On
1	<b>Regions</b>	Nothing -- create these first.
2	<b>Entities</b>	A Region must exist before you can create an Entity.
3	<b>Projects</b>	An Entity must exist before you can create a Project.

If you attempt to create an Entity without first creating a Region, you will have no Region to select in the form. The same applies to Projects and Entities.

---

## Regions

Regions are the top level of the hierarchy. They typically represent geographic areas or business divisions.

### Creating a Region

1. Navigate to **Org Structure** in the left sidebar and click **Regions**.
2. Click the "**New**" button in the top-right corner.
3. Fill in the fields:
  - **Name** -- the display name for the region (e.g. "Middle East & North Africa").
  - **Code** -- a short, unique identifier (e.g. "MENA", "EMEA", "APAC"). This code is used in CSV imports, reports, and system filters. It does not appear on contracts themselves.
  - **Description** -- an optional note explaining the scope or purpose of the region.
4. Click **Save**.

### Region Fields

Field	Required	Description
<b>Name</b>	Yes	Display name shown throughout the application.
<b>Code</b>	Yes	Unique short code used in imports, reports, and filters.
<b>Description</b>	No	Free-text description of the region's scope.

---

## Entities

Entities represent legal or organisational units within a Region -- companies, subsidiaries, or divisions.

### Creating an Entity

1. Navigate to **Org Structure** in the left sidebar and click **Entities**.

2. Click the **"New"** button.
3. Fill in the fields:
  - o **Name** -- the display name (e.g. "Digittal AE").
  - o **Code** -- a unique short code (e.g. "DGT-AE"). Used in the same contexts as the Region code.
  - o **Legal Name** -- the full legal name as it appears on official registration documents.
  - o **Registration Number** -- the company registration or incorporation number.
  - o **Region** -- select the Region this entity belongs to (dropdown populated from existing Regions).
  - o **Parent Entity** -- optionally select another Entity as the parent, creating a hierarchical relationship (for example, a holding company as the parent and its subsidiary as the child).
4. Click **Save**.

### Entity Fields

Field	Required	Description
Name	Yes	Display name shown throughout the application.
Code	Yes	Unique short code used in imports, reports, and filters.
Legal Name	No	Full legal name as registered.
Registration Number	No	Company registration / incorporation number.
Region	Yes	The Region this entity belongs to.
Parent Entity	No	Another Entity that serves as this entity's parent in the hierarchy.

### Entity Hierarchy (Parent/Child Relationships)

The **Parent Entity** field allows you to model corporate structures beyond the three-level Region/Entity/Project chain. For example:

Region: MENA

- └ Entity: Digittal Holdings (parent)
  - └ Entity: Digittal AE (child – parent = Digittal Holdings)
  - └ Entity: Digittal SA (child – parent = Digittal Holdings)
    - └ Project: PRJ-010

This is useful when a holding company has multiple operating subsidiaries, each with its own projects and contracts. The parent-child relationship is displayed in the **Organization Visualization** page (see below).

### Assigning Jurisdictions to an Entity

Entities can operate in one or more legal jurisdictions. To associate jurisdictions with an entity:

1. Open the Entity record.
2. Navigate to the **Jurisdictions** tab (relation manager).
3. Click **"Attach"** and select one or more Jurisdiction records.

Jurisdictions assigned to an entity are used in compliance tracking and regulatory reporting. See the Jurisdictions section below for details on creating Jurisdiction records.

---

## Projects

Projects are the most granular level of the hierarchy. Each project belongs to exactly one Entity and serves as the organisational bucket to which contracts are assigned.

### Creating a Project

1. Navigate to **Org Structure** in the left sidebar and click **Projects**.

2. Click the **"New"** button.
3. Fill in the fields:
  - o **Name** -- the project's display name.
  - o **Code** -- a unique short code (e.g. "PRJ-001").
  - o **Entity** -- select the Entity this project belongs to (dropdown populated from existing Entities).
  - o **Description** -- an optional description of the project's scope or purpose.
4. Click **Save**.

**Project Fields**

Field	Required	Description
Name	Yes	Display name shown throughout the application.
Code	Yes	Unique short code used in imports, reports, and filters.
Entity	Yes	The Entity this project belongs to.
Description	No	Free-text description of the project.

---

## Jurisdictions

Jurisdictions represent legal territories and regulatory environments. They are assigned to Entities to indicate where those entities operate.

**Creating a Jurisdiction**

1. Navigate to **Administration** in the left sidebar and click **Jurisdictions**.
2. Click the **"New"** button.
3. Fill in the fields:
  - o **Name** -- the jurisdiction's display name (e.g. "United Arab Emirates", "United Kingdom").
  - o **Country Code** -- the ISO country code (e.g. "AE", "GB", "US").
  - o **Regulatory Body** -- the name of the primary regulatory body in this jurisdiction (e.g. "Securities and Commodities Authority", "Financial Conduct Authority").
  - o **Description** -- optional additional context about the jurisdiction's regulatory environment.
4. Click **Save**.

**Jurisdiction Fields**

Field	Required	Description
Name	Yes	Display name of the jurisdiction.
Country Code	Yes	ISO country code.
Regulatory Body	No	Name of the primary regulatory authority.
Description	No	Free-text description of the regulatory context.

**How Jurisdictions Are Used**

- **Entity compliance** -- when an Entity has one or more Jurisdictions attached, the system can track regulatory compliance requirements specific to those territories.
- **Contract context** -- contracts created under an Entity inherit awareness of the Entity's jurisdictions, which informs compliance checks and reporting.

---

## Signing Authorities

Signing Authorities define who is authorised to sign contracts and up to what value. They provide a critical governance control -- ensuring that contracts above a certain value require sign-off from appropriately senior personnel.

## Creating a Signing Authority

1. Navigate to **Administration** in the left sidebar and click **Signing Authorities**.
2. Click the **"New"** button.
3. Fill in the fields:
  - **User** -- select the user who is being granted signing authority.
  - **Entity** -- optionally select an Entity to scope this authority to a specific entity. Leave blank for broader authority.
  - **Project** -- optionally select a Project to scope this authority to a specific project. Leave blank for entity-wide or global authority.
  - **Authority Level** -- the level of signing authority (used to differentiate tiers of authorisation).
  - **Max Contract Value** -- the maximum contract value (in the specified currency) that this user is authorised to sign.
  - **Currency** -- the currency for the max contract value threshold.
4. Click **Save**.

## Signing Authority Fields

Field	Required	Description
<b>User</b>	Yes	The user being granted signing authority.
<b>Entity</b>	No	Scopes the authority to a specific entity.
<b>Project</b>	No	Scopes the authority to a specific project.
<b>Authority Level</b>	Yes	The tier of signing authority.
<b>Max Contract Value</b>	Yes	Maximum value this user can authorise.
<b>Currency</b>	Yes	Currency for the max contract value.

## Scoping Rules

Signing authority can be scoped at different levels of specificity:

Entity	Project	Scope
Set	Set	Authority applies only to the specified project within the specified entity.
Set	Blank	Authority applies to all projects within the specified entity.
Blank	Blank	Authority applies across the entire organisation (global).

When a contract is submitted for signing, CCRS checks whether the proposed signer has a Signing Authority record that covers the contract's entity (or project) and that the contract's value does not exceed the signer's **Max Contract Value** threshold.

---

## Organization Visualization

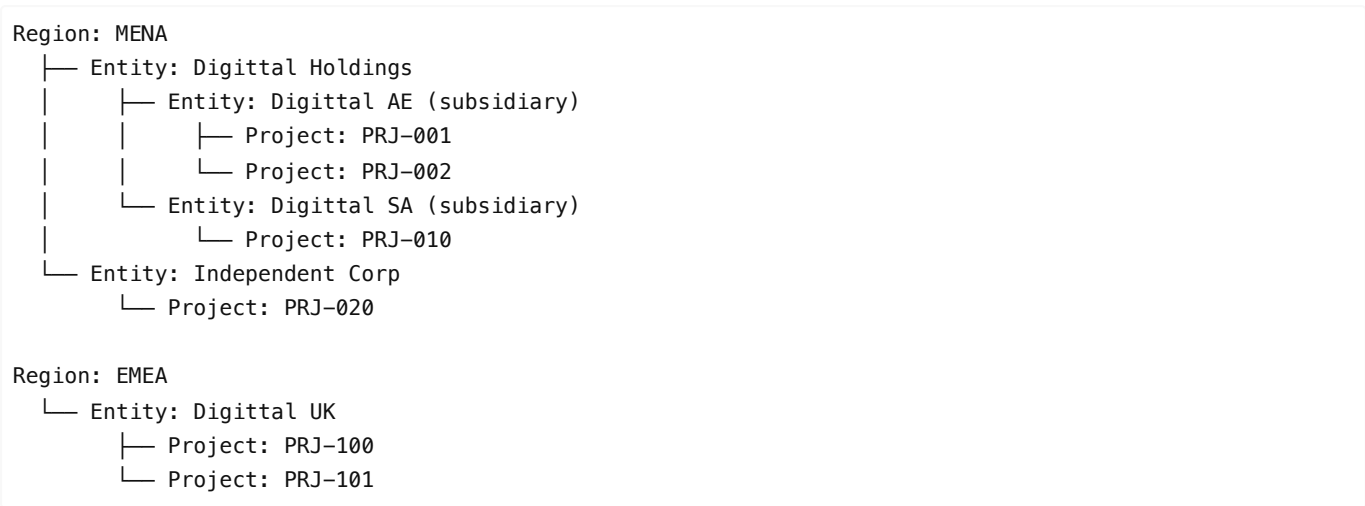
The **Organization Visualization** page provides a visual, tree-based display of your complete organisational structure.

### Accessing the Visualization

1. Navigate to **Org Structure** in the left sidebar and click **Organization Visualization**.
2. The page renders a hierarchical tree showing the full Region, Entity, and Project structure.

### What the Visualization Shows

The tree displays the complete hierarchy:



- **Regions** are the top-level nodes.
- **Entities** are nested under their Region, with parent-child Entity relationships shown as sub-nesting.
- **Projects** are the leaf nodes, nested under their Entity.

Use Cases

- **Onboarding** -- new team members can quickly understand the organisational topology and where their projects sit.
- **Planning** -- when creating workflow templates scoped to specific parts of the hierarchy, the visualization helps confirm the correct Region/Entity/Project target.
- **Audit** -- auditors can verify the organisational structure and confirm that the hierarchy reflects the actual corporate structure.

Permissions

Only **System Admin** users can manage organisational structure. The table below summarises access.

Action	System Admin	Legal	Commercial	Finance	Operations	Audit
Create / edit Regions	Yes	--	--	--	--	--
Create / edit Entities	Yes	--	--	--	--	--
Create / edit Projects	Yes	--	--	--	--	--
Create / edit Jurisdictions	Yes	--	--	--	--	--
Create / edit Signing Authorities	Yes	--	--	--	--	--
View Organization Visualization	Yes	Yes	Yes	Yes	Yes	Yes

All users can view the Organization Visualization page, but only System Admins can modify the underlying records.

Best Practices

- **Establish Regions first.** Before onboarding any contracts, define your full set of Regions. Changing Regions later is possible but affects all downstream Entities and their contracts.
- **Use consistent Code conventions.** Adopt a naming scheme for codes (e.g. region codes in uppercase like "MENA", entity codes with a prefix like "DGT-AE", project codes with a sequential pattern like "PRJ-001") and document it so all administrators follow the same convention.
- **Model your real corporate structure.** Use the Parent Entity field to reflect your actual holding-company / subsidiary relationships. This makes the Organization Visualization accurate and useful for auditors.

- **Assign Jurisdictions early.** Attaching Jurisdictions to Entities as soon as they are created ensures that compliance tracking is active from the first contract.
  - **Set conservative signing authority limits.** Start with lower Max Contract Value thresholds and increase them as needed. It is easier to grant additional authority than to revoke it after a contract has been signed.
  - **Review the visualization after changes.** After adding or restructuring Entities, visit the Organization Visualization page to confirm the hierarchy looks correct before creating contracts against the new structure.
- 

## 12. Vendor Portal

### What is the Vendor Portal?

The Vendor Portal is an external-facing interface for counterparty contacts -- vendors, suppliers, and partners who need to interact with CCRS without having a full internal account. It provides a simplified, secure way to view contracts assigned to your organisation, upload documents when requested, and receive notifications from CCRS users.

The portal uses magic-link authentication, meaning there are no passwords to remember. Access is controlled entirely through verified email addresses linked to a counterparty record in CCRS.

---

### Accessing the Vendor Portal

Your account manager will provide you with the vendor portal URL. The login process works as follows:

1. Visit the vendor portal login page at `/vendor/login`.
2. Enter your registered email address and click **Send Login Link**.
3. If your email is recognised, a magic link is sent to your inbox. The link is valid for **24 hours**.
4. Open the email and click the link to access the portal. A secure session is created automatically.
5. If your email is not recognised, you will see an error message. Contact your account manager to be added to the system.

No password is required at any point. The magic link is your secure credential.

```
flowchart TD
    A[Vendor Visits /vendor/login] --> B[Enters Email Address]
    B --> C{Email Recognized?}
    C -->|Yes| D[Magic Link Sent to Email]
    C -->|No| E[Error: Contact Your Account Manager]
    D --> F[Vendor Clicks Link in Email]
    F --> G[Token Verified & Session Created]
    G --> H[Vendor Dashboard]
    H --> I{Choose Action}
    I -->|View Contracts| J[Browse Assigned Contracts]
    I -->|Upload Documents| K[Submit Documents for Review]
    I -->|View Notifications| L[Read Notification Messages]
    I -->|Logout| M[Session Ended]
    J --> N[Download Contract Files]
```

---

### Vendor Dashboard

After logging in, you arrive at the vendor dashboard. The dashboard provides an overview of your activity and quick access to the three main areas of the portal.

#### Your Contracts

A list of contracts associated with your counterparty organisation. Each entry shows the contract title, type, current status, and key dates (start date, end date, renewal date). Click any contract to view its details and download the contract file (PDF or DOCX).

## Documents

A list of documents you have uploaded through the portal, along with their status. This section shows the document title, the contract it is associated with, and the upload date.

## Notifications

Messages sent to you by CCRS users regarding your contracts. Each notification includes a title and message body. Unread notifications are highlighted. Click a notification to mark it as read.

---

## Viewing Contracts

From the dashboard, click **Your Contracts** to browse all contracts assigned to your counterparty.

- **Browse** -- contracts are listed with their title, type, status, and key dates.
- **View details** -- click a contract to see its full details, including contract terms, parties, and timeline.
- **Download files** -- download the contract document in PDF or DOCX format directly from the contract detail page.

You can only see contracts that are associated with your counterparty. Contracts belonging to other organisations are not visible.

---

## Uploading Documents

CCRS users may request that you upload documents -- for example, compliance certificates, insurance documentation, or signed counterparts. To upload a document:

1. Navigate to the contract that requires the document, or click **Upload Document** from the documents section.
2. Click **Upload Document**.
3. Select the file from your device. Accepted formats are PDF and DOCX.
4. Enter a descriptive **document title** (e.g., "Certificate of Insurance 2026").
5. Click **Submit**.

After submission:

- The document is stored securely and associated with the relevant contract.
  - CCRS users who manage that contract are notified of the new upload.
  - The document appears in your **Documents** list with its upload date.
- 

## Notifications

CCRS users can send notifications to vendor portal users. Common notification scenarios include:

- A new contract has been assigned to your counterparty.
- A document upload has been requested.
- A contract status has changed (e.g., approved, renewed, or terminated).
- A reminder about an upcoming deadline.

Notifications appear on the dashboard and in the dedicated **Notifications** section. Unread notifications are highlighted. Click a notification to view the full message and mark it as read.

---

## Security

The vendor portal is designed with multiple layers of security to protect your data and your organisation's contracts.

### Magic Link Authentication

- No passwords are stored or transmitted. Authentication is handled entirely through single-use magic links.
- Tokens are generated using a **cryptographically secure pseudo-random number generator (CSPRNG)**.

- Only the **SHA-256 hash** of each token is stored in the database. The plaintext token appears only in the email link. Even if the database were compromised, tokens could not be extracted.
- Tokens expire after **24 hours**. After expiry, you must request a new login link.

## Session Security

- All sessions are time-limited. You will be logged out automatically after the session expires.
- Logging out invalidates the session immediately.

## Data Isolation

- You can only access contracts associated with your counterparty. Contracts belonging to other organisations are not visible.
- Uploaded documents are stored securely in S3-compatible object storage and are not publicly accessible.

---

## For CCRS Administrators

This section is for internal CCRS users who manage vendor portal access.

### Managing Vendor Users

Navigate to **Administration --> Vendor Users** to manage vendor portal accounts.

Action	How
Create a vendor user	Click <b>New Vendor User</b> . Enter the user's name, email address, and select the linked counterparty.
Edit a vendor user	Click the vendor user's name to edit their details (name, email, counterparty).
Activate / deactivate	Toggle the <b>Active</b> switch to enable or disable portal access. Deactivated users cannot log in.
View vendor users by counterparty	Filter the vendor user list by counterparty to see all portal users for a specific organisation.

### Vendor User Fields

Field	Description
Name	The vendor contact's full name.
Email	The email address used for magic link login. Must be unique across all vendor users.
Counterparty	The counterparty organisation this vendor user belongs to. Determines which contracts they can see.
Phone	Optional contact phone number.
Active	Whether the user can log in to the vendor portal.

### Sending Notifications to Vendors

CCRS users can send notifications to vendor portal users through the VendorNotificationService. Notifications are delivered to the vendor's portal dashboard -- they appear the next time the vendor logs in or refreshes their dashboard.

### Best Practices

- **One counterparty per vendor user** -- each vendor user is linked to exactly one counterparty. If a contact works across multiple counterparties, create separate vendor user accounts for each.

- **Deactivate rather than delete** -- when a vendor contact leaves their organisation, deactivate their account rather than deleting it. This preserves the audit trail of their document uploads and portal activity.
  - **Review active vendor users periodically** -- ensure that only current contacts have active portal access.
- 

## 13. External Signing Guide

This guide is for people outside of Digittal Group who have been asked to sign a document. You do not need a CCRS account, a password, or any special software. Everything happens in your web browser.

---

### You Have Received a Signing Request

You will receive an email from CCRS containing a signing invitation. The email includes:

- The **title** of the contract you are being asked to sign.
- The **name** of the person who sent the request.
- A **"Sign Document"** button that takes you directly to the signing page.

Click the button to open the signing page in your browser. No account or login is needed -- the link itself is your secure access to the document.

The link is valid for **7 days**. If it expires before you sign, ask the sender to resend the invitation.

---

### Viewing the Document

When you open the signing page, the contract is displayed in a built-in PDF viewer. Take the time to read through the entire document before signing.

- **Scroll through the document** to review all pages.
- If the sender has enabled **page-by-page viewing**, you must scroll through every page before the signing controls become available. A progress bar at the top of the page shows how many pages you have viewed (e.g., "Pages viewed: 3 of 12").
- If **page initials** are required, each page will display an **"Initial"** button. Click the button on each page to draw your initials in the small canvas that appears. The progress indicator shows how many pages you have initialed (e.g., "Pages initialed: 5 of 12").

You cannot skip ahead -- the system tracks which pages you have viewed and initialed. The signing controls remain disabled until all requirements are met.

---

### Signing the Document

Once you have reviewed the document (and initialed pages, if required), the signature area becomes active. You can choose from four methods to provide your signature.

#### Draw

Use your mouse, trackpad, or finger (on a touchscreen device) to draw your signature on the canvas. If you make a mistake, click **Clear** and try again.

#### Type

Type your full name into the text field. The system renders it as a signature-style image. This is the fastest method.

#### Upload

Upload an existing image of your signature. The file must be a **PNG** or **JPEG** image. This is useful if you have a scanned copy of your handwritten signature saved on your device.

## Camera

Use your device's camera to photograph your handwritten signature on a piece of paper:

1. When prompted, click **Allow** to grant camera access.
2. A live camera preview appears on screen.
3. Write your signature on a plain piece of white paper.
4. Hold the paper up to the camera so your signature is clearly visible.
5. Click **Capture**.
6. The system processes the image automatically -- it removes the paper background and isolates your signature.
7. Review the result. If you are satisfied, click **Accept**. If not, click **Retake** to try again.

Camera signing requires a modern browser and an HTTPS connection. If the camera does not activate, check that you have granted permission in your browser settings.

---

## Using a Saved Signature

If you have signed documents through CCRS before and chose to save your signature, your stored signatures will appear in a "**Saved Signatures**" section on the signing page. Click any saved signature to select it immediately, without needing to draw, type, upload, or capture a new one.

If you do not have any saved signatures, this section will not appear.

---

## Submitting Your Signature

After choosing or creating your signature:

1. Review the signature preview to make sure it looks correct.
2. Click **Submit Signature**.
3. You will be asked whether you want to **save this signature for future use**. If you check this option, the next time you are asked to sign a document through CCRS, your signature will be available for one-click selection.
4. A confirmation message is displayed on screen.
5. You will receive a **confirmation email** once your signature has been recorded.

All parties involved in the signing process will be notified as the process progresses.

---

## Declining to Sign

If you are unable or unwilling to sign the document, you can decline:

1. Click **Decline** on the signing page.
2. Enter a **reason** for declining. This field is required.
3. Click **Confirm Decline**.

What happens after you decline:

- The sender is notified of your decision and can read the reason you provided.
  - Your status is recorded as "declined" in the signing record.
  - Declining is **final** for this signing session. You cannot change your mind and sign after declining. If circumstances change, the sender must create a new signing session.
- 

## What Happens After You Sign

Once you submit your signature, the following steps occur automatically:

1. **Your signature is recorded** with a secure timestamp, your IP address, and the method you used.
2. **Other signers are notified** (if applicable):

- In **sequential** signing, the next signer in the order receives their invitation email.
- In **parallel** signing, other signers may have already received their links and can sign independently.

3. **When all signers have signed**, the system finalises the document:

- All signatures are embedded into the final PDF at their designated positions.
- An **audit certificate** is generated -- a separate document that records who signed, when, from which IP address, and the method used.
- A **SHA-256 hash** of the completed document is computed for tamper detection.

4. **You receive a completion email** confirming that the signing process is finished and the document has been finalised.

## Troubleshooting

Issue	Solution
<b>The link does not work</b>	The link may have expired. Signing links are valid for 7 days. Ask the sender to resend the invitation.
<b>Camera is not working</b>	Ensure you are using HTTPS (the address bar should show a padlock icon). Check that you have granted camera permission in your browser settings. Try a different browser if the issue persists.
<b>Cannot submit my signature</b>	Make sure you have viewed all required pages. If page initials are required, check that you have initialed every page. The progress indicators at the top of the page show your status.
<b>Page says "already signed"</b>	You have already submitted your signature for this document. No further action is needed.
<b>Page says "session expired"</b>	The signing session has expired. Contact the sender to request a new signing session.
<b>Browser compatibility</b>	Use a modern, up-to-date browser: Google Chrome, Mozilla Firefox, Apple Safari, or Microsoft Edge. Mobile browsers on iOS and Android are supported. Internet Explorer is not supported.
<b>Page loads slowly or PDF does not appear</b>	Check your internet connection. Try refreshing the page. If the problem continues, try a different browser or device.
<b>Accidentally closed the page</b>	You can re-open the signing link from your email as long as it has not expired and you have not already signed or declined.

## Privacy and Security

Your security is important. Here is how the signing system protects you:

- **No account required** -- you do not need to create an account, set a password, or install any software.
- **Secure tokens** -- the link in your email contains a unique, cryptographically generated token. Only the hash of this token is stored in the system -- the link itself is the only place the full token exists.
- **Encrypted connection** -- all communication between your browser and the signing server uses HTTPS encryption.
- **Time-limited access** -- signing links expire after 7 days and sessions expire after 30 days.
- **Audit trail** -- every action you take on the signing page (viewing the document, signing, declining) is recorded with a timestamp and your IP address for legal accountability.
- **Document integrity** -- the document is hashed before and after signing. Any tampering with the document after signing would be detectable.

## Frequently Asked Questions

**Do I need to install anything?** No. Everything runs in your web browser. No plugins, extensions, or desktop software are required.

**Can I sign on my phone or tablet?** Yes. The signing page works on mobile devices. The Draw method works well with touchscreens.

**What if I need to sign multiple documents?** Each document has its own signing link. You will receive a separate email for each document that requires your signature.

**Is my signature legally binding?** Electronic signatures collected through CCRS are intended to be legally binding. The system records your identity, intent to sign, the timestamp, and the document content to support enforceability. Consult your legal advisor if you have specific questions about the legal validity of electronic signatures in your jurisdiction.

**Can the sender see my signature after I submit it?** Yes. Your signature is embedded in the final signed PDF document, which is accessible to all parties involved in the contract.

**What if I have questions about the document content?** Contact the person who sent you the signing request. Their name and organisation are included in the invitation email.

**Can I download a copy of the signed document?** Once all parties have signed and the document is finalised, you will receive a completion email. Depending on the sender's configuration, this email may include a link to download the final signed document.

## 14. Role Reference Matrix

CCRS uses role-based access control (RBAC) powered by Spatie Permissions to govern what every user can see and do across the platform. Each user is assigned exactly one of six roles, and that role determines the full set of permissions available to them. This chapter provides a comprehensive reference for every role-permission combination, explains how restricted contracts and signing authorities add secondary access checks, and includes a visual decision tree to help you understand how the system evaluates access requests.

### The Six Roles

CCRS defines six roles. Each serves a distinct function within the contract lifecycle.

Role	Purpose
System Admin	Full platform access. Manages users, organisation structure, workflow templates, signing authorities, bulk operations, and all configuration. Receives wildcard (*) permissions.
Legal	Primary contract lifecycle role. Creates, edits, and manages contracts, counterparties, KYC templates, and wiki contracts. Views audit logs and approves override requests. Accesses escalation and compliance features.
Commercial	Focuses on contract origination and merchant agreements. Creates contracts and counterparties, generates merchant agreements, and submits override requests for legal review.
Finance	Read-only access to contracts with full access to financial reporting, the analytics dashboard, and the AI cost report.
Operations	Read-only access to contracts with day-to-day operational tools: key date tracking and reminder management.
Audit	Read-only access to contracts and audit logs for independent review. Can view and export reports.

### Full Permissions Grid

The table below lists every major feature and action in CCRS and shows which roles have access. "Yes" indicates the role can perform the action; a dash (-) indicates the action is not available to that role.

Feature / Action	System Admin	Legal	Commercial	Finance	Operations	Audit
<b>Contracts</b>						
View Contracts	Yes	Yes	Yes	Yes	Yes	Yes
Create Contracts	Yes	Yes	Yes	-	-	-
Edit Contracts	Yes	Yes	-	-	-	-
Delete Contracts	Yes	-	-	-	-	-
Restrict / Unrestrict Contracts	Yes	Yes	-	-	-	-
<b>Counterparties</b>						
View Counterparties	Yes	Yes	Yes	-	-	-
Create Counterparties	Yes	Yes	Yes	-	-	-
Edit Counterparties	Yes	Yes	-	-	-	-
Merge Counterparties	Yes	-	-	-	-	-
<b>Override Requests</b>						
Submit Override Requests	-	-	Yes	-	-	-
Approve / Reject Overrides	Yes	Yes	-	-	-	-
<b>Organisation &amp; Configuration</b>						
Manage Workflow Templates	Yes	-	-	-	-	-
Manage Org Structure	Yes	-	-	-	-	-
Manage KYC Templates	Yes	Yes	-	-	-	-
Manage Signing Authorities	Yes	-	-	-	-	-
Manage Vendor Users	Yes	-	-	-	-	-
<b>Bulk Operations</b>						
Bulk Data Upload	Yes	-	-	-	-	-
Bulk Contract Upload	Yes	-	-	-	-	-
<b>Audit &amp; Compliance</b>						
View Audit Logs	Yes	Yes	-	-	-	Yes
View Reports	Yes	Yes	-	Yes	-	Yes
Export Reports	Yes	Yes	-	Yes	-	Yes
Analytics Dashboard	Yes	-	-	Yes	-	-
AI Cost Report	Yes	-	-	Yes	-	-
<b>AI &amp; Analysis</b>						
Trigger AI Analysis	Yes	Yes	-	-	-	-

Start Redline Review	Yes	Yes	-	-	-	-
<b>Wiki Contracts</b>						
Manage Wiki Contracts	Yes	Yes	-	-	-	-
<b>Signing</b>						
Send for Signing	Yes	Yes	Yes	-	-	-
<b>Escalations &amp; Dates</b>						
View Escalations	Yes	Yes	-	-	-	-
View Key Dates	Yes	Yes	Yes	-	Yes	-
Manage Reminders	Yes	Yes	Yes	-	Yes	-
<b>Merchant Agreements</b>						
View Merchant Agreements	Yes	Yes	Yes	-	-	-
Generate Merchant Agreements	Yes	Yes	Yes	-	-	-
<b>Personal</b>						
Notification Preferences	Yes	Yes	Yes	Yes	Yes	Yes
My Signatures	Yes	Yes	Yes	Yes	Yes	Yes
Help & Guide	Yes	Yes	Yes	Yes	Yes	Yes

## Role-Based Access Decision Tree

The following diagram illustrates how CCRS evaluates whether a user can access a given resource. The system first checks the user's role, then the resource type, and finally whether the contract is restricted.

flowchart TD

A[User Requests Access to Resource] --> B{What is User's Role?}

B -->|system\_admin| C[Full Access to All Resources]

B -->|legal| D{Resource Type?}

B -->|commercial| E{Resource Type?}

B -->|finance| F{Resource Type?}

B -->|operations| G{Resource Type?}

B -->|audit| H{Resource Type?}

D -->|Contracts| D1[View + Create + Edit]

D -->|Counterparties| D2[View + Create + Edit]

D -->|KYC| D3[View + Manage]

D -->|Audit Logs| D4[View Only]

D -->|Wiki Contracts| D5[View + Create + Edit]

D -->|Override Requests| D6[View + Approve/Reject]

E -->|Contracts| E1[View + Create]

E -->|Counterparties| E2[View + Create]

E -->|Merchant Agreements| E3[View + Create + Generate]

E -->|Override Requests| E4[Submit Only]

F -->|Contracts| F1[View Only]

```

F -->|Reports| F2[View + Export]
F -->|Analytics| F3[Full Dashboard Access]

G -->|Contracts| G1[View Only]
G -->|Key Dates| G2[View + Manage Reminders]

H -->|Contracts| H1[View Only]
H -->|Audit Logs| H2[View Only]
H -->|Reports| H3[View + Export]

D1 --> I{Is Contract Restricted?}
E1 --> I
F1 --> I
G1 --> I
H1 --> I
I -->|No| J[Access Granted]
I -->|Yes| K{User in Authorized List?}
K -->|Yes| J
K -->|No| L[Access Denied – Contact Admin]

```

## How to Read the Decision Tree

1. **Role check** -- CCRS first identifies the user's role. System Admins bypass all further checks and receive full access.
2. **Resource type check** -- for every other role, the system determines the type of resource being requested and matches it against the role's permission set.
3. **Restricted contract check** -- if the resource is a contract (or a record linked to a contract) and that contract is flagged as restricted, CCRS performs a secondary check against the contract's authorised user list (see below).

## Restricted Contracts

Any contract can be flagged as **restricted** by a System Admin or Legal user. When a contract is restricted, standard role-based permissions are necessary but not sufficient -- the user must also appear on the contract's authorised access list.

### How Restriction Works

1. A System Admin or Legal user opens the contract and enables the **Is Restricted** toggle.
2. The system creates a `ContractUserAccess` record for the users who should retain access.
3. From that point forward, any user who attempts to view or interact with the contract is checked against two criteria:
  - **Role permission** -- does their role grant access to contracts at all?
  - **Authorised list** -- is their user ID present in the `ContractUserAccess` table for this contract?
4. Both conditions must be met. A user with the Legal role who is not on the authorised list will be denied access to that specific contract.

### Managing the Authorised List

- Navigate to the contract's detail page.
- Open the **Access Control** tab (visible to System Admins and Legal users).
- Add or remove users from the authorised list.
- Changes take effect immediately.

### Who Can Restrict Contracts

Action	System Admin	Legal	Commercial	Finance	Operations	Audit
Flag a contract as restricted	Yes	Yes	-	-	-	-
Add users to the authorised list	Yes	Yes	-	-	-	-

Remove users from the authorised list	Yes	Yes	-	-	-	-
---------------------------------------	-----	-----	---	---	---	---

## Use Cases

- **Sensitive M&A contracts** that should only be visible to the deal team.
- **Executive compensation agreements** restricted to HR and legal counsel.
- **Regulatory matters** where access must be limited to avoid conflicts of interest.

## Signing Authorities

Signing Authorities govern **who is authorised to sign contracts** on behalf of a given entity or project, and up to what monetary value. This is a separate layer from role-based access control -- having the permission to "send for signing" does not automatically grant the authority to sign.

## How Signing Authorities Work

Each Signing Authority record defines:

Field	Description
<b>User</b>	The individual granted signing authority.
<b>Entity</b>	The legal entity on whose behalf the user may sign.
<b>Project</b>	(Optional) A specific project within the entity. If set, the authority is scoped to contracts linked to that project only.
<b>Value Limit</b>	The maximum contract value the user may sign. Contracts with a total value exceeding this limit require a higher authority.

## Enforcement

When a user attempts to sign a contract, CCRS checks:

1. **Does a Signing Authority record exist** for this user, matching the contract's entity (and project, if applicable)?
2. **Does the contract's total value fall within** the user's value limit?

If either check fails, the signing action is blocked and the user is informed that they do not have sufficient signing authority. This prevents unauthorised commitments and enforces the organisation's delegation-of-authority matrix.

## Managing Signing Authorities

Only **System Admins** can create, edit, or delete Signing Authority records.

1. Navigate to **Signing Authorities** in the left sidebar.
2. Click **"New Signing Authority"** to create a record, or click an existing record to edit it.
3. Select the **User**, **Entity**, and optionally a **Project**.
4. Set the **Value Limit** in the contract currency.
5. Click **Save**.

## Example Scenario

User	Entity	Project	Value Limit
Jane Smith	Digital Holdings	--	50,000
Jane Smith	Digital Holdings	Project Alpha	200,000
David Lee	Digital Holdings	--	500,000

In this example:

- Jane Smith can sign any Digittal Holdings contract up to 50,000 -- except for Project Alpha contracts, where her limit is 200,000.
  - David Lee can sign any Digittal Holdings contract up to 500,000, regardless of project.
  - A contract worth 600,000 would require a different signatory with a higher limit.
- 

## Quick Reference by Role

The following summaries provide a fast lookup for each role.

### System Admin

Full access to every feature and setting. System Admins are responsible for platform configuration, user management, organisation structure, workflow templates, signing authorities, bulk operations, and vendor user management. They can view and manage all contracts, including restricted ones.

### Legal

The primary contract management role. Legal users create, edit, and manage contracts and counterparties; handle KYC templates; manage wiki contracts; approve or reject override requests; trigger AI analysis and redline reviews; view audit logs; and access escalations. They can restrict contracts and manage authorised access lists.

### Commercial

Focused on contract origination and merchant operations. Commercial users create contracts and counterparties, generate merchant agreements, submit override requests for legal review, send contracts for signing, and manage key dates and reminders. They cannot edit existing contracts or approve override requests.

### Finance

A reporting-oriented role. Finance users have read-only access to contracts and full access to the reports suite, the analytics dashboard, and the AI cost report. They manage their own notification preferences and signatures but do not create or modify contract records.

### Operations

An operational support role. Operations users have read-only access to contracts and focus on key date tracking and reminder management to ensure contractual obligations are met on time. They do not access reports, audit logs, or configuration settings.

### Audit

An independent review role. Audit users have read-only access to contracts and audit logs, plus the ability to view and export reports. This role is designed for internal audit teams who need to verify compliance without the ability to modify any records.

---

## Best Practices

- **Assign the least-privilege role.** Give each user the role that matches their job function. Avoid assigning System Admin to users who only need Legal or Commercial access.
  - **Use restricted contracts for sensitive deals.** Do not rely solely on role-based access for confidential contracts -- flag them as restricted and maintain a tight authorised user list.
  - **Review signing authorities regularly.** When employees change roles or leave the organisation, update or revoke their signing authority records promptly.
  - **Audit role assignments periodically.** System Admins should review the user list at least quarterly to confirm that role assignments still reflect current responsibilities.
  - **Document your delegation-of-authority matrix.** Maintain a clear record of which roles and value limits apply to each entity and project, and ensure signing authority records in CCRS match the approved matrix.
-

# 15. Compliance & Audit

CCRS provides a comprehensive compliance and audit framework designed to give your organisation full visibility into every action taken on the platform. From automatic audit logging of every record change, through a dedicated signing audit trail with legal-grade evidence, to regulatory framework tracking and document integrity verification -- every layer is built to support internal governance, external audits, and legal defensibility.

## Audit Logging

Every create, update, and delete operation in CCRS is automatically recorded in the audit log. This happens transparently -- users do not need to take any action to generate audit entries, and no user can disable or bypass the logging system.

### What Gets Logged

Every time a record is created, updated, or deleted anywhere in CCRS, the `AuditService` captures a complete snapshot of the change.

Field	Description
User	The user who performed the action, identified by name and user ID.
Event	The type of operation: created, updated, or deleted.
Record Type	The type of record affected (e.g. Contract, Counterparty, Workflow Stage, Signing Session).
Record ID	The unique identifier of the specific record.
Old Values	For updates and deletes, the field values before the change. Stored as structured JSON.
New Values	For creates and updates, the field values after the change. Stored as structured JSON.
IP Address	The IP address from which the action was performed.
User Agent	The browser and operating system used to perform the action.
Timestamp	The exact date and time of the action (UTC).

### Immutability

Audit log entries are **immutable**. Once written, they cannot be edited, overwritten, or deleted -- not even by System Admins. This guarantees the integrity of the audit trail and ensures it can serve as reliable evidence during internal reviews, external audits, or legal proceedings.

### Viewing Audit Logs

Audit logs are accessible to three roles:

Role	Access Level
System Admin	Full access to all audit log entries across the entire platform.
Legal	View access to all audit log entries.
Audit	View access to all audit log entries.

To view audit logs:

1. Navigate to **Audit Logs** in the left sidebar.
2. The page displays a chronological list of all logged events, newest first.

- 3. Each entry shows the user, event type, record type, and timestamp in the list view.
- 4. Click any entry to expand it and view the full detail -- including old values, new values, IP address, and user agent.

Filtering Audit Logs

Use the filters at the top of the Audit Logs page to narrow results:

- **User** -- show only actions performed by a specific user.
- **Event Type** -- filter by created, updated, or deleted.
- **Record Type** -- focus on a specific model (e.g. only Contract changes, only Counterparty changes).
- **Date Range** -- restrict to a specific time window.
- **Record ID** -- search for all changes to a specific record.

Contract-Level Audit History

In addition to the centralised Audit Logs page, each contract's detail page includes an **Activity** tab that shows the audit history for that specific contract. This provides a focused view of every change made to a single contract over its lifetime without the need to filter the global log.

Signing Audit Trail

The signing process has its own dedicated audit trail, separate from the general audit log. The `SigningAuditLog` captures every event in the lifecycle of a signing session with the detail required for legal defensibility.

Events Tracked

Event	Description
Session Created	A new signing session was initiated for a contract. Records who created the session and when.
Invitation Sent	A signing invitation (magic link) was dispatched to a signer. Records the recipient and delivery channel.
Document Viewed	A signer opened the document for review. Proves the signer had access to the full document before signing.
Signature Submitted	A signer applied their signature (typed, drawn, uploaded, or webcam). Records the signature method used.
Signing Declined	A signer explicitly declined to sign. Records the reason if provided.
Session Completed	All required signatures have been collected and the session is marked as complete.
Session Cancelled	The session was cancelled before completion. Records who cancelled and the reason.
Reminder Sent	A follow-up reminder was sent to a signer who has not yet signed.

Data Captured Per Event

Each signing audit log entry includes:

Field	Description
Signing Session	The session this event belongs to.
Signer	The name and identifier of the signer involved (if applicable).
Event Type	One of the events listed above.

<b>IP Address</b>	The IP address from which the event occurred.
<b>User Agent</b>	The browser and operating system used.
<b>Timestamp</b>	The exact date and time of the event (UTC).
<b>Additional Data</b>	Event-specific metadata -- e.g. signature method for "Signature Submitted", decline reason for "Signing Declined".

## Viewing the Signing Audit Trail

The signing audit trail is accessible from two locations:

1. **Signing Session Detail Page** -- navigate to a signing session record and open the **Audit Trail** tab to see every event for that session in chronological order.
2. **Contract Detail Page** -- the contract's **Signing** tab shows all signing sessions for that contract, each with a link to its full audit trail.

## Contract Immutability

CCRS enforces strict immutability rules on contracts that have reached certain lifecycle stages. This ensures that the terms of an executed agreement cannot be altered after the fact.

### Immutability Rules

Contract Status	Editable?	Explanation
Draft	Yes	Contracts in draft status can be freely edited by users with the appropriate role permissions.
In Review	Yes	Contracts under review can still be modified as part of the negotiation and approval process.
Approved	Limited	Approved contracts can be modified only through specific override workflows.
Executed	No	Once a contract is executed (all signatures collected and the signing session is complete), all fields become <b>read-only</b> . No user, including System Admins, can edit the contract record directly.
Archived	No	Archived contracts are permanently read-only. They are retained for reference and audit purposes.

## Making Changes to Executed Contracts

When business circumstances require changes to an executed contract, CCRS supports three mechanisms -- each of which creates a new, linked contract record rather than modifying the original:

- **Amendment** -- a formal modification to specific terms of the executed contract. The amendment is linked to the parent contract and goes through its own workflow and signing process.
- **Renewal** -- an extension or replacement of the contract for a new term. The renewal references the original contract and captures updated terms.
- **Side Letter** -- a supplementary agreement that modifies or clarifies specific provisions without rewriting the full contract.

In all three cases, the original executed contract remains unchanged. The linked record provides a clear audit trail showing what was modified, when, and by whom.

## Document Integrity

CCRS uses SHA-256 cryptographic hashing to guarantee that contract documents are not tampered with during the signing process.

## How It Works

1. **At session creation** -- when a signing session is initiated, CCRS computes a SHA-256 hash of the original contract PDF and stores it in the signing session record. This hash acts as a digital fingerprint of the document as it existed at the moment signing began.
2. **At session completion** -- when all signatures have been collected and the session is marked as complete, CCRS computes a SHA-256 hash of the final signed PDF and stores it alongside the original hash.
3. **Verification** -- at any point after signing, the stored hashes can be compared against the actual document files. If the hash of the file on disk matches the stored hash, the document is confirmed to be unaltered. If the hashes do not match, the document has been modified since the hash was recorded.

## What SHA-256 Guarantees

- **Tamper detection** -- any change to the document, no matter how small (even a single byte), produces a completely different hash value. This makes undetected tampering computationally infeasible.
- **Non-repudiation** -- the stored hash proves that the document the signers reviewed and signed is identical to the document on file.
- **Chain of custody** -- the pair of hashes (pre-signing and post-signing) establishes that the document was consistent throughout the signing process.

## Where Hashes Are Stored

Document hashes are stored in the signing session record and included in the audit certificate (see below). They are also recorded in the signing audit trail, providing multiple independent references.

---

## Audit Certificate

When a signing session completes, CCRS automatically generates an **audit certificate** -- a standalone PDF that serves as a comprehensive record of the entire signing process.

### What the Audit Certificate Contains

Section	Contents
<b>Contract Details</b>	Contract title, reference number, parties, effective date, total value.
<b>Signing Session</b>	Session ID, creation date, completion date, total duration.
<b>Signer Information</b>	For each signer: full name, email address, role, signature method (typed/drawn/uploaded/webcam), signing timestamp.
<b>IP Addresses</b>	The IP address from which each signer accessed and signed the document.
<b>User Agents</b>	The browser and operating system each signer used.
<b>Document Hashes</b>	The SHA-256 hash of the original document (at session creation) and the SHA-256 hash of the signed document (at session completion).
<b>Event Timeline</b>	A chronological list of every signing event (created, sent, viewed, signed, completed) with timestamps.

## Storage

The audit certificate is stored as a separate PDF file alongside the signed contract document. Both files are retained in the same secure storage location (S3 with server-side encryption).

## Accessing the Audit Certificate

1. Navigate to the contract's detail page.
2. Open the **Signing** tab.
3. Click on the completed signing session.
4. The audit certificate is available as a downloadable PDF in the session detail view.

## Legal Significance

The audit certificate provides the evidence required to demonstrate in a legal proceeding that:

- The signers were properly identified and invited.
  - Each signer reviewed the document before signing.
  - The document was not altered between invitation and signing.
  - Each signature was captured with a specific method, from a specific IP address, at a specific time.
  - The complete chain of events is recorded and verifiable.
- 

## Regulatory Frameworks

CCRS allows your organisation to define regulatory frameworks and check contracts against them for compliance. This feature is designed for organisations that operate across multiple jurisdictions or industries with varying regulatory requirements.

### Defining a Regulatory Framework

System Admins and Legal users can create and manage regulatory frameworks.

1. Navigate to **Regulatory Frameworks** in the left sidebar.
2. Click **"New Regulatory Framework"**.
3. Complete the following fields:

Field	Description
<b>Name</b>	A descriptive name for the framework (e.g. "GDPR Data Processing Requirements", "South Africa Consumer Protection Act").
<b>Description</b>	A detailed explanation of what the framework covers and when it applies.
<b>Jurisdiction</b>	The jurisdiction this framework applies to (selected from the Jurisdictions list).
<b>Requirements</b>	A structured list of specific requirements that contracts must meet. Stored as JSON and displayed as a checklist during compliance checks.
<b>Is Active</b>	Whether this framework is currently in use. Inactive frameworks are retained for reference but are not included in compliance checks.

4. Click **Save**.

### Running a Compliance Check

The `RegulatoryComplianceService` evaluates contracts against applicable regulatory frameworks.

1. Open a contract's detail page.
2. Navigate to the **Compliance** tab.
3. Click **"Run Compliance Check"**.
4. CCRS evaluates the contract against all active regulatory frameworks that match the contract's jurisdiction and type.
5. The results appear as a list of **Compliance Findings**.

### Compliance Findings

Each finding represents a specific compliance observation or issue discovered during the check.

Field	Description
<b>Regulatory Framework</b>	The framework that generated this finding.
<b>Finding Type</b>	The category of the finding (e.g. missing clause, non-compliant term, documentation gap).
<b>Severity</b>	The seriousness of the finding: <b>Critical, High, Medium, or Low</b> .
<b>Description</b>	A detailed explanation of the issue and what needs to be addressed.
<b>Status</b>	The current resolution status (see below).
<b>Remediation</b>	Notes on how the finding was or should be resolved.

## Finding Statuses

Compliance findings progress through four statuses:

Status	Meaning
<b>Open</b>	The finding has been identified but no action has been taken yet.
<b>In Progress</b>	Remediation is underway -- someone is actively working to resolve the issue.
<b>Resolved</b>	The issue has been addressed and the contract is now compliant with this requirement.
<b>Waived</b>	The finding has been reviewed and a decision has been made to accept the risk without remediation. This should be used sparingly and with documented justification.

## Updating Finding Status

1. Open the contract's **Compliance** tab.
2. Click on the finding you want to update.
3. Change the **Status** field to the appropriate value.
4. If resolving or waiving, add a note in the **Remediation** field explaining the action taken and the rationale.
5. Click **Save**.

All status changes are recorded in the audit log, providing a full history of how each finding was handled.

## Compliance Monitoring

CCRS provides multiple views for monitoring compliance across your contract portfolio.

### Analytics Dashboard -- Compliance Widget

The **Analytics Dashboard** (accessible to System Admins and Finance users) includes a **Compliance Overview** widget that displays:

- **Total findings** across all contracts, broken down by severity.
- **Open vs. resolved** findings as a percentage and trend over time.
- **Frameworks with the most findings** -- helping identify which regulatory areas require the most attention.
- **Average time to resolution** -- how long it takes your organisation to address compliance findings.

### Contract Detail -- Compliance Tab

Each contract's detail page includes a **Compliance** tab that shows:

- All compliance findings for that specific contract.
- The status of each finding (open, in progress, resolved, waived).

- The regulatory framework each finding relates to.
- A summary of the contract's overall compliance posture.

## Reports

The **Reports** section (accessible to System Admins, Legal, Finance, and Audit roles) includes compliance-related reports that can be filtered by:

- **Jurisdiction** -- focus on a specific regulatory jurisdiction.
- **Framework** -- filter by a specific regulatory framework.
- **Severity** -- show only critical or high-severity findings.
- **Status** -- filter by open findings only, or show the full history.
- **Date range** -- restrict to findings created or resolved within a specific period.

Reports can be exported to Excel or PDF for distribution to stakeholders, regulatory bodies, or external auditors.

---

## Data Retention

CCRS retains audit and compliance data according to the following policies:

Data Type	Retention Policy
<b>Audit Logs</b>	Retained indefinitely. Audit log entries are never deleted or purged.
<b>Signing Audit Logs</b>	Retained indefinitely alongside the associated signing sessions.
<b>Audit Certificates</b>	Retained indefinitely as PDF files in secure storage.
<b>Contract Documents</b>	Retained indefinitely in S3 with server-side encryption.
<b>Compliance Findings</b>	Retained indefinitely, including resolved and waived findings.
<b>Regulatory Frameworks</b>	Retained indefinitely. Inactive frameworks are preserved for historical reference.

## Storage Security

- All files (contract documents, signed PDFs, audit certificates) are stored in S3 with server-side encryption enabled.
- Access to stored files is controlled through the same RBAC and restricted-contract mechanisms described in [Chapter 14 -- Role Reference Matrix](#).
- Database records (audit logs, signing audit logs, compliance findings) are stored in the MySQL database with access restricted to authenticated users with appropriate role permissions.

---

## Preparing for an External Audit

When your organisation undergoes an external audit or regulatory review, CCRS provides the tools to respond efficiently.

### Step-by-Step Guide

1. **Identify the scope.** Determine which contracts, time periods, and regulatory frameworks the audit covers.
2. **Export audit logs.** Navigate to **Audit Logs**, apply the appropriate date range and record type filters, and export the results. The export includes all fields: user, event, record type, old/new values, IP address, and timestamps.
3. **Export compliance findings.** Navigate to **Reports**, select the compliance report, filter by the relevant jurisdiction and framework, and export to Excel or PDF.
4. **Gather signing evidence.** For each contract in scope, download the **audit certificate** from the signing session detail page. This single document contains the complete signing chain of custody.

5. **Verify document integrity.** If the auditor requires proof that contract documents have not been tampered with, provide the SHA-256 hashes from the signing session record and demonstrate that recomputing the hash of the stored file produces the same value.
  6. **Provide role and access documentation.** Export the user list with role assignments to demonstrate that access controls are in place and appropriate.
- 

## Best Practices

- **Do not treat compliance findings as optional.** Every open finding represents a risk. Establish a target resolution time and track against it.
- **Use the "Waived" status sparingly.** Waiving a finding means accepting the risk. Always document the rationale and ensure a senior stakeholder has approved the decision.
- **Review regulatory frameworks regularly.** Regulations change. Schedule a quarterly review of your active frameworks to ensure requirements are current.
- **Download audit certificates promptly.** While certificates are retained indefinitely in CCRS, keeping local copies in your organisation's document management system provides an additional layer of redundancy.
- **Leverage the Analytics Dashboard.** The Compliance Overview widget gives you a real-time snapshot of your organisation's compliance posture. Review it weekly to catch emerging issues early.
- **Restrict access to audit logs appropriately.** Audit log access is limited to System Admin, Legal, and Audit roles by design. Do not create workarounds that expose audit data to other roles -- the principle of least privilege protects the integrity of the audit trail.
- **Train your team on immutability.** Ensure all users understand that executed contracts cannot be edited and that changes must be made through amendments, renewals, or side letters. This prevents confusion and support requests.