Documento Técnico-Funcional

Módulo de Registro (Registry)

Sistema de Gestión de Correspondencia

Versión	1.0
Fecha	26/05/2026
Clasificación	Interno – Uso técnico
Área responsable	Dirección Nacional de Derecho de Autor (DNDA)
Componente	Registry API · Frontend Angular
Stack tecnológico	.NET 9 · PostgreSQL · Kafka · Angular 17+

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

1. Propósito y Alcance

Este documento describe de manera técnico-funcional el funcionamiento completo del Módulo de Registro del Sistema de Gestión de Correspondencia (SGC) de la Dirección Nacional de Derecho de Autor (DNDA). Abarca la arquitectura del microservicio backend (Registry API), el modelo de dominio, los flujos de negocio, los endpoints REST, la integración con Kafka, el modelo de base de datos y el componente frontend Angular que interactúa con este módulo.

1.1 Objetivo del Módulo

El Módulo de Registro gestiona el ciclo de vida completo de las salidas de correspondencia generadas por la Oficina de Registro de la DNDA. A partir de los radicados de entrada recibidos del sistema CorrespondenceManager, el módulo permite: asignar responsables, gestionar documentos y evidencias, registrar notificaciones, aplicar decisiones (registro, devolución, negación), controlar el periodo de ejecutoria y cerrar formalmente el trámite.

1.2 Alcance

- Microservicio Registry API (.NET 9, arquitectura hexagonal)
- Entidades de dominio y reglas de negocio
- Endpoints REST y contratos de datos (DTOs)
- Integración con Apache Kafka (producción y consumo de eventos)
- Modelo de base de datos relacional (PostgreSQL)
- Worker de sincronización de radicados desde CorrespondenceManager
- Componente frontend Angular (RegistryWorkbenchPage) y servicio de API

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

2. Arquitectura del Microservicio

2.1 Visión General

Registry API es un microservicio independiente construido con .NET 9 y ASP.NET Core. Sigue una arquitectura hexagonal (Ports & Adapters) organizada en cuatro proyectos de capa:

Diagrama de Capas — Arquitectura Hexagonal

API Layer	**`Registry.API`**	Controladores REST, Workers Kafka, DI Config
Application Layer	**`Registry.Application`**	IRegistryCaseService, DTOs, Puertos (Interfaces)
Domain Layer	**`Registry.Domain`**	RegistryCase, Enums, Reglas de Negocio
Infrastructure Layer	**`Registry.Infrastructure`**	RegistryDbContext, RegistryCaseRepository (EF Core + PostgreSQL)

Capa	Proyecto	Responsabilidad
Domain	Registry.Domain	Entidades, enumeraciones y reglas de dominio puras. Sin dependencias externas.
Application	Registry.Application	Interfaces (puertos), DTOs, servicios de aplicación con la lógica de negocio.
Infrastructure	Registry.Infrastructure	Implementación de repositorios, DbContext de Entity Framework Core con PostgreSQL.
API	Registry.API	Controladores REST, workers (Background Services), configuración de DI y pipeline HTTP.

2.2 Inyección de Dependencias

El archivo Program.cs registra los componentes principales:

```
services.AddDbContext<RegistryDbContext>(UseNpgsql(...))
services.AddScoped<IRegistryCaseRepository, RegistryCaseRepository>()
services.AddScoped<IRegistryCaseService, RegistryCaseService>()
services.AddHostedService<RegistryCorrespondenceSyncWorker>()
services.AddDndaKafka(configuration)   // Kafka producer compartido
```

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

2.3 Configuración y Variables de Entorno

Variable / Sección	Tipo	Descripción
ConnectionStrings__BDConnection	String	Cadena de conexión PostgreSQL.
Kafka:BootstrapServers	String	Servidor(es) Kafka para producción y consumo.
Kafka:RegistryEventsTopic	String	Topic de publicación RegistryCaseCreatedEvent.
Kafka:CorrespondenceEventsTopic	String	Topic consumido por el worker de sincronización.
RegistryCorrespondenceSync:Enabled	Bool	Habilita/deshabilita el worker de sincronización.
RegistryCorrespondenceSync:GroupId	String	Consumer group ID de Kafka.
RegistryCorrespondenceSync:DependencyIds	Int[]	IDs de dependencias de Registro.
RegistryCorrespondenceSync:DependencyNameContains	String	Texto a buscar en el nombre de dependencia.
Cors:AllowedOrigins	String[]	Orígenes permitidos para CORS.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

3. Modelo de Dominio

3.1 Entidad Principal: RegistryCase

RegistryCase es el agregado raíz que representa un trámite de registro. Contiene toda la información del caso y sus colecciones de sub-entidades:

Campo	Tipo	Nullable	Descripción
Id	Guid	No	Identificador único del trámite (PK).
FilingNumber	string(80)	No	Número de radicado. Único en la tabla (UNIQUE INDEX).
ProcessType	enum	No	Tipo de proceso: PetitionRight, Modification, Cancellation, RegistryRequest.
Status	enum	No	Estado actual del trámite (ver sección 3.2).
Subject	string(300)	No	Asunto o descripción breve del trámite.
CategoryCode	string(80)	Sí	Código de clasificación de respuesta.
CategoryName	string(200)	Sí	Nombre de la clasificación.
OfficeCode	string(80)	No	Código numérico de la dependencia destino.
OfficeName	string(200)	No	Nombre de la dependencia.
ApplicantName	string(200)	Sí	Nombre del solicitante.
ApplicantEmail	string(320)	Sí	Correo del solicitante.
ResponsibleUserId	string(150)	Sí	ID del usuario responsable asignado.
ResponsibleName	string(200)	Sí	Nombre del usuario responsable.
AssignedByUserId	string(150)	Sí	ID del usuario que realizó la asignación.
AssignedAt	DateTime?	Sí	Fecha y hora de la asignación.
FiledAt	DateTime	No	Fecha de radicación original.
CreatedAt	DateTime	No	Fecha de creación del registro en la BD.
UpdatedAt	DateTime?	Sí	Última actualización.
DueAt	DateTime?	Sí	Fecha límite de respuesta.
ExecutoryDeadlineAt	DateTime?	Sí	Fecha límite del periodo de ejecutoria (FiledAt + 10 días).
HasRequiredForm	bool	No	Formulario requerido recibido.
HasNotificationEvidence	bool	No	Evidencia de notificación registrada.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

HasRetentionEvidence	bool	No	Evidencia TRD registrada.
HasCorrespondenceOutput	bool	No	Número de salida registrado.
HasResourceFiled	bool	No	Interesado presentó recurso en ejecutoria.
CorrespondenceOutputNumber	string(100)	Sí	Número radicado de salida (formato 2-AAAA-N).
ClosedAt	DateTime?	Sí	Fecha de cierre del trámite.
CloseReason	string(1000)	Sí	Motivo del cierre.

3.2 Estados del Trámite (RegistryStatus)

El estado del trámite controla las transiciones permitidas:

Diagrama de Máquina de Estados

1. Filed	2. AssignedToOffice	3. AssignedToResponsible
4. InTechnicalReview	5. InChiefReview	6. Registered ✓
7. Returned ↩	8. Rejected ✗	9. Notified
10. InExecutoryPeriod	11. Closed ■	12. ReturnedForCorrections

Flujo de Transiciones

```
── Creación / Worker Kafka ──────────────────────────────────────
  → AssignedToOffice (estado inicial de todo trámite)

── Asignación ───────────────────────────────────────────────────
  AssignedToOffice → InTechnicalReview    (RegistryRequest)
  AssignedToOffice → AssignedToResponsible (otros tipos)

── Revisión de Documentos ───────────────────────────────────────
  InTechnicalReview / AssignedToResponsible → InChiefReview
  InChiefReview → ReturnedForCorrections  (rechazado por jefe)

── Decisión Final ───────────────────────────────────────────────
  InTechnicalReview → Registered | Returned | Rejected

── Notificación ─────────────────────────────────────────────────
  Cualquier estado → Notified

── Ejecutoria (Modificación / Cancelación) ──────────────────────
  Notified → InExecutoryPeriod  (10 días desde notificación)
```

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

```
── Cierre ──────────────────────────────────────────────────────────
  Cualquier estado (cumpliendo prerrequisitos) → Closed
```

Valor	Nombre	Descripción
1	Filed	Radicado. Estado inicial cuando se crea manualmente sin asignación inmediata.
2	AssignedToOffice	Asignado a la Oficina de Registro. Estado inicial estándar al crear o sincronizar desde Kafka.
3	AssignedToResponsible	Asignado a un funcionario responsable (procesos diferentes a RegistryRequest).
4	InTechnicalReview	En revisión técnica (aplica a RegistryRequest).
5	InChiefReview	Enviado a revisión del jefe de oficina.
6	Registered	Registrado. Decisión final positiva para solicitudes de registro.
7	Returned	Devuelto al solicitante.
8	Rejected	Negado.
9	Notified	Notificado al interesado.
10	InExecutoryPeriod	En periodo de ejecutoria (Modificación/Cancelación tras notificación).
11	Closed	Cerrado. Estado terminal.
12	ReturnedForCorrections	Devuelto para correcciones (cuando el jefe rechaza un documento).

3.3 Tipos de Proceso (RegistryProcessType)

Valor	Nombre	Descripción
1	PetitionRight	Derecho de petición. Requiere notificación. Sin periodo de ejecutoria.
2	Modification	Modificación de registro. Requiere formulario, notificación y ejecutoria (10 días).
3	Cancellation	Cancelación de registro. Mismos requisitos que Modificación.
4	RegistryRequest	Solicitud de registro. Flujo de revisión técnica/jefe y decisión formal.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

3.4 Sub-Entidades del Dominio

3.4.1 RegistryDocument — Documentos y Evidencias

Campo	Tipo	Descripción
DocumentType	enum	Tipo: PetitionResponse, AdministrativeAct, RemissionLetter, RegistryCertificate, ReturnLetter, RejectionResolution, NotificationEvidence, RetentionTableEvidence, CorrectionVersion.
Name	string(260)	Nombre del documento o archivo.
StorageReference	string(1000)	Referencia de almacenamiento (URL, path, ID externo).
Version	int	Versión; se incrementa automáticamente por tipo+nombre.
RequiresChiefReview	bool	Si true, al adjuntar se cambia el estado a InChiefReview.
IsApproved	bool	true si fue aprobado en la revisión del jefe.
ReviewedByUserId	string(150)	ID del revisor.
ReviewComment	string(1000)	Comentario de la revisión.

3.4.2 RegistryNotification — Notificaciones

Campo	Tipo	Descripción
Channel	enum	Canal: DataMessage, CertifiedMail, InPerson.
Recipient	string(320)	Destinatario de la notificación.
EvidenceReference	string(1000)	Referencia de la evidencia del envío.
Notes	string(1000)	Observaciones adicionales.
SentByUserId	string(150)	ID del usuario que registró la notificación.
SentAt	DateTime	Fecha y hora de la notificación.

3.4.3 RegistryAuditEntry — Auditoría

Cada operación que modifica el trámite genera un RegistryAuditEntry. Tipos de acción:

Created(1) · AssignedToOffice(2) · AssignedToResponsible(3) · TechnicalReviewStarted(4) · SentToChiefReview(5) · CorrectionRequested(6) · DocumentApproved(7) · Registered(8) · Returned(9) · Rejected(10) · Notified(11) · ExecutoryPeriodStarted(12) · ResourceFiled(13) · DocumentArchived(14) · CorrespondenceOutputRegistered(15) · Closed(16) · BulkAnnotationRegistered(17) · DocumentRegistered(18)

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

3.4.4 RegistryBulkAnnotation — Anotaciones Masivas

Permite registrar anotaciones técnicas que afectan múltiples registros simultáneamente. Campos: BatchReference, AnnotationText y AffectedRecords.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

4. API REST — Endpoints

Base URL: api/v1/registry-cases | Formato: JSON | Autenticación: JWT Bearer Token

Todos los endpoints devuelven ResultModel<T> con propiedades data, hasError y messages. Códigos HTTP: 200 (OK), 201 (Created), 400 (validación), 404 (no encontrado), 409 (conflicto).

4.1 Búsqueda de Trámites

GET `/api/v1/registry-cases`

Busca y filtra trámites con paginación. Devuelve lista de RegistryCaseSummaryDto.

Parámetros / Body

Campo	Tipo	Descripción
ProcessType	string?	Filtro por tipo de proceso.
Status	string?	Filtro por estado actual del trámite.
CategoryCode	string?	Filtro por código de clasificación.
ResponsibleUserId	string?	Filtro por ID de responsable.
Search	string?	Búsqueda libre en FilingNumber, Subject, etc.
Page	int	Página (default: 1).
PageSize	int	Tamaño de página (default: 50).

Respuesta: RegistryCaseSummaryDto[] — ordenados por FiledAt descendente.

4.2 Detalle del Trámite

GET `/api/v1/registry-cases/{id}`

Devuelve el detalle completo incluyendo documentos, notificaciones, auditoría y anotaciones.

Parámetros / Body

Campo	Tipo	Descripción
id	Guid	Identificador del trámite.

Respuesta: RegistryCaseDetailDto — detalle completo con todas las colecciones.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

4.3 Crear Trámite

POST `/api/v1/registry-cases`

Crea un nuevo trámite. Valida: FilingNumber único (409), Modification/Cancellation requieren HasRequiredForm=true, RegistryRequest requiere CategoryCode. Estado inicial: AssignedToOffice. Publica RegistryCaseCreatedEvent en Kafka.

Parámetros / Body

Campo	Tipo	Descripción
FilingNumber*	string(80)	Número de radicado único.
ProcessType*	enum	Tipo de proceso.
Subject*	string(300)	Asunto del trámite.
CategoryCode	string(80)	Requerido si ProcessType = RegistryRequest.
CategoryName	string(200)	Nombre de la categoría.
ApplicantName	string(200)	Nombre del solicitante.
ApplicantEmail	string(320)	Correo del solicitante.
FiledAt	DateTime?	Fecha de radicación (default: ahora).
DueAt	DateTime?	Fecha límite de respuesta.
HasRequiredForm	bool	Formulario recibido.
CreatedByUserId*	string(150)	ID del usuario creador.

Respuesta: RegistryCaseDetailDto (HTTP 201).

4.4 Asignar Responsable

POST `/api/v1/registry-cases/{id}/assign`

Asigna un funcionario. Estado → InTechnicalReview si es RegistryRequest, o AssignedToResponsible en otros casos.

Parámetros / Body

Campo	Tipo	Descripción
ResponsibleUserId*	string(150)	ID del nuevo responsable.
ResponsibleName*	string(200)	Nombre del responsable.
AssignedByUserId*	string(150)	ID del asignador.
AssignedByName	string(200)	Nombre del asignador.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

Notes	string(1000)	Observaciones.

Respuesta: RegistryCaseDetailDto actualizado.

4.5 Adjuntar Documento / Evidencia

POST `/api/v1/registry-cases/{id}/documents`

Adjunta un documento. Si ya existe mismo tipo+nombre, incrementa versión. Si RequiresChiefReview=true → InChiefReview.

Parámetros / Body

Campo	Tipo	Descripción
DocumentType*	enum	Tipo de documento.
Name*	string(260)	Nombre del documento.
StorageReference	string(1000)	Referencia de almacenamiento.
RequiresChiefReview	bool	Eleva estado a InChiefReview.
IsRetentionEvidence	bool	Marca HasRetentionEvidence=true en el caso.
CreatedByUserId*	string(150)	ID del usuario.

Respuesta: RegistryCaseDetailDto actualizado.

4.6 Revisar Documento (Jefe)

POST `/api/v1/registry-cases/{id}/documents/review`

El jefe aprueba o rechaza. Si Approved=false → ReturnedForCorrections.

Parámetros / Body

Campo	Tipo	Descripción
DocumentId*	Guid	ID del documento.
Approved*	bool	true = aprobado, false = devuelto.
ReviewedByUserId*	string(150)	ID del revisor.
ReviewComment	string(1000)	Comentario.

Respuesta: RegistryCaseDetailDto actualizado.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

4.7 Registrar Notificación

POST `/api/v1/registry-cases/{id}/notifications`

Registra la notificación al interesado. Activa HasNotificationEvidence=true. Estado → Notified.

Parámetros / Body

Campo	Tipo	Descripción
Channel*	enum	Canal: DataMessage, CertifiedMail, InPerson.
Recipient*	string(320)	Destinatario.
EvidenceReference	string(1000)	Referencia de evidencia.
Notes	string(1000)	Observaciones.
SentByUserId*	string(150)	ID del usuario notificador.

Respuesta: RegistryCaseDetailDto (estado = Notified).

4.8 Resolución de Solicitud de Registro

POST `/api/v1/registry-cases/{id}/registry-decision`

Aplica decisión final (solo RegistryRequest). TargetStatus: Registered, Returned o Rejected.

Parámetros / Body

Campo	Tipo	Descripción
TargetStatus*	enum	Decisión: Registered, Returned o Rejected.
PerformedByUserId*	string(150)	ID del usuario decisor.
Notes	string(1000)	Sustento de la decisión.

Respuesta: RegistryCaseDetailDto con nuevo estado.

4.9 Iniciar Periodo de Ejecutoria

POST `/api/v1/registry-cases/{id}/executory-period`

Inicia ejecutoria para Modificación o Cancelación. Requiere HasNotificationEvidence=true. ExecutoryDeadlineAt = NotifiedAt + 10 días. Estado → InExecutoryPeriod.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

Parámetros / Body

Campo	Tipo	Descripción
NotifiedAt	DateTime?	Fecha de notificación (default: hoy).
PerformedByUserId*	string(150)	ID del usuario.

Respuesta: RegistryCaseDetailDto (estado = InExecutoryPeriod).

4.10 Registrar Recurso

POST `/api/v1/registry-cases/{id}/resources`

Marca si el interesado presentó recurso en periodo de ejecutoria.

Parámetros / Body

Campo	Tipo	Descripción
HasResourceFiled*	bool	true si presentó recurso.
PerformedByUserId*	string(150)	ID del usuario.
Notes	string(1000)	Observaciones.

Respuesta: RegistryCaseDetailDto actualizado.

4.11 Registrar Número de Salida

POST `/api/v1/registry-cases/{id}/correspondence-output`

Registra el número de radicado de salida. Valida formato 2-AAAA-N y unicidad. Activa HasCorrespondenceOutput=true.

Parámetros / Body

Campo	Tipo	Descripción
OutputNumber*	string(100)	Formato: 2-AAAA-consecutivo.
PerformedByUserId*	string(150)	ID del usuario.
Notes	string(1000)	Observaciones.

Respuesta: RegistryCaseDetailDto (HasCorrespondenceOutput=true).

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

4.12 Registrar Anotación Masiva

POST `/api/v1/registry-cases/{id}/bulk-annotations`

Agrega anotación técnica para procesos por lotes.

Parámetros / Body

Campo	Tipo	Descripción
BatchReference*	string(120)	Referencia del lote.
AnnotationText*	text	Texto de la anotación.
AffectedRecords*	int	Número de registros afectados (mín. 1).
CreatedByUserId*	string(150)	ID del usuario.

Respuesta: RegistryCaseDetailDto actualizado.

4.13 Cerrar Trámite

POST `/api/v1/registry-cases/{id}/close`

Cierra el trámite verificando todos los prerrequisitos. Estado final: Closed.

Parámetros / Body

Campo	Tipo	Descripción
ClosedByUserId*	string(150)	ID del usuario que cierra.
CloseReason	string(1000)	Motivo del cierre.

Respuesta: RegistryCaseDetailDto con estado Closed y ClosedAt.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

5. Reglas de Negocio y Flujos

5.1 Flujo General del Trámite

Un trámite de registro sigue el siguiente ciclo de vida general:

Paso	Descripción
1. Recepción / Creación	Trámite creado con estado AssignedToOffice (manual vía API o automático por worker Kafka).
2. Asignación de Responsable	Jefe de Registro asigna funcionario → InTechnicalReview (RegistryRequest) o AssignedToResponsible.
3. Revisión Técnica	El responsable adjunta documentos. Si alguno requiere jefe → InChiefReview.
4. Revisión del Jefe	El jefe aprueba o rechaza el documento. Si rechaza → ReturnedForCorrections.
5. Decisión / Resolución	RegistryRequest: aplica Registered, Returned o Rejected. Otros: se procede a notificación.
6. Notificación	Se registra la notificación formal. Estado → Notified.
7. Ejecutoria (Mod/Cancel)	Periodo de ejecutoria de 10 días. Estado → InExecutoryPeriod. El interesado puede presentar recurso.
8. Registro de Salida	Se registra el radicado de salida formato 2-AAAA-N.
9. Cierre	Se verifican prerrequisitos y el trámite se cierra → Closed.

5.2 Máquina de Estados

Transiciones de estado controladas por la capa Application:

Estado Origen	Estado Destino	Acción que dispara la transición
— (nuevo)	AssignedToOffice	CreateAsync / Worker sync
AssignedToOffice	AssignedToResponsible	AssignAsync (no RegistryRequest)
AssignedToOffice	InTechnicalReview	AssignAsync (RegistryRequest)
AssignedToResponsible	InChiefReview	AddDocumentAsync (RequiresChiefReview=true)
InTechnicalReview	InChiefReview	AddDocumentAsync (RequiresChiefReview=true)

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

InChiefReview	ReturnedForCorrections	ReviewDocumentAsync (Approved=false)
InTechnicalReview / InChiefReview	Registered | Returned | Rejected	ResolveRegistryRequestAsync
Cualquiera	Notified	RegisterNotificationAsync
Notified	InExecutoryPeriod	StartExecutoryPeriodAsync (Mod/Cancel)
Cualquiera	Closed	CloseAsync (cumpliendo prerrequisitos)

5.3 Validaciones por Tipo de Proceso

PetitionRight

- No requiere formulario.
- Requiere notificación para cierre.
- No tiene periodo de ejecutoria.

Modification / Cancellation

- HasRequiredForm debe ser true en la creación.
- Requiere notificación.
- Requiere inicio de periodo de ejecutoria (10 días desde notificación).
- El cierre requiere que la ejecutoria haya vencido O que el interesado haya presentado recurso.

RegistryRequest

- Requiere CategoryCode en la creación.
- El flujo pasa por InTechnicalReview e InChiefReview.
- Se debe aplicar una decisión formal (Registered / Returned / Rejected).
- Si la decisión fue Registered, no requiere notificación para cerrar.
- No tiene periodo de ejecutoria.

5.4 Validaciones del Número de Salida

El número de radicado de salida debe cumplir el formato 2-AAAA-N (regex: ^2-\d{4}-\d+$). Además, debe ser único en toda la base de datos de Registry.

5.5 Prerrequisitos de Cierre

CloseAsync verifica que se cumplan TODOS los siguientes prerrequisitos:

Prerrequisito	Descripción
HasRetentionEvidence = true	Se debe haber adjuntado al menos una evidencia TRD.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

HasCorrespondenceOutput = true	Se debe haber registrado el número de salida del gestor.
HasNotificationEvidence = true	Requerido para PetitionRight, Modification, Cancellation. Excepción: RegistryRequest con decisión Registered.
Ejecutoria vencida o recurso	Para InExecutoryPeriod: ExecutoryDeadlineAt <= ahora, O HasResourceFiled = true.

5.6 Versionamiento de Documentos

Cuando se adjunta un documento con el mismo DocumentType y Name que uno ya existente, el sistema incrementa automáticamente el campo Version.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

6. Integración con Apache Kafka

Diagrama de Flujo Kafka

Componente	Acción	Topic	Evento / Descripción
CorrespondenceManager	PUBLICA	`CorrespondenceEventsTopic`	CorrespondenceRadicatedEvent / CorrespondenceAssignedToRegistryEvent
RegistryCorrespondenceSyncWorker	CONSUME	`CorrespondenceEventsTopic`	Procesa, crea/actualiza RegistryCase en PostgreSQL
RegistryCaseService	PUBLICA	`RegistryEventsTopic`	RegistryCaseCreatedEvent (al crear un trámite vía API)
Otros Microservicios	CONSUMEN	`RegistryEventsTopic`	Reaccionan ante nuevos trámites de registro

6.1 Evento Producido: RegistryCaseCreatedEvent

Al crear un trámite, RegistryCaseService publica IntegrationEvent<RegistryCaseCreatedEvent> en Kafka:RegistryEventsTopic, usando FilingNumber como message key.

```
Topic:    Kafka:RegistryEventsTopic (ej. "registry-events")
Key:      FilingNumber del trámite creado
EventType: "RegistryCaseCreatedEvent"
Payload:  { Id, FilingNumber, ProcessType, Status, Subject, ... }
```

6.2 Worker de Sincronización: RegistryCorrespondenceSyncWorker

BackgroundService que consume CorrespondenceEventsTopic publicado por CorrespondenceManager. Crea automáticamente trámites de registro cuando un radicado es asignado a la dependencia de Registro.

Eventos consumidos

Evento	Descripción
CorrespondenceRadicatedEvent	Radicado creado en CorrespondenceManager. Si la dependencia corresponde a Registro, se crea o actualiza el RegistryCase.
CorrespondenceAssignedToRegistryEvent	Radicado trasladado a la Oficina de Registro. Misma lógica.

Lógica de procesamiento (UpsertRegistryCaseAsync)

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

- Verifica si la dependencia destino corresponde a Registro (por DependencyId o DependencyName).
- Si ya existe RegistryCase con ese FilingNumber: actualiza OfficeCode, OfficeName, ResponsibleUserId, DueAt. Registra auditoría AssignedToOffice.
- Si no existe: crea nuevo RegistryCase con estado AssignedToOffice y resuelve ProcessType.
- Dead-letter: DbUpdateException y errores inesperados confirman offset Kafka para no bloquear el consumer.

Resolución de ProcessType desde nombre

```
Contiene "modific" → Modification
Contiene "cancel"  → Cancellation
Contiene "derecho" o "petici" → PetitionRight
Otro valor          → RegistryRequest (default)
```

Configuración del Consumer

```
AutoOffsetReset:   Earliest
EnableAutoCommit:  false   (commit manual tras procesamiento exitoso)
GroupId:           Kafka:RegistryCorrespondenceSync:GroupId
```

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

7. Modelo de Base de Datos

Registry utiliza PostgreSQL con Entity Framework Core 9. El contexto RegistryDbContext gestiona las 5 tablas del módulo. Los enumerados se almacenan como strings para facilitar la legibilidad.

7.1 Tablas

Tabla: registry_cases

PK: id (UUID) — Tabla principal. Un registro por trámite.

- ux_registry_cases_filing_number (UNIQUE)
- ix_registry_cases_status
- ix_registry_cases_process_type
- ix_registry_cases_category_code
- ix_registry_cases_responsible_user_id

Tabla: registry_documents

PK: id (UUID) — Documentos adjuntos. FK: registry_case_id → registry_cases.id (CASCADE DELETE).

- ix_registry_documents_case_id

Tabla: registry_notifications

PK: id (UUID) — Notificaciones registradas. FK: registry_case_id → registry_cases.id (CASCADE DELETE).

Tabla: registry_audit_entries

PK: id (UUID) — Auditoría de acciones. FK: registry_case_id → registry_cases.id (CASCADE DELETE).

- ix_registry_audit_entries_case_id

Tabla: registry_bulk_annotations

PK: id (UUID) — Anotaciones masivas. FK: registry_case_id → registry_cases.id (CASCADE DELETE).

7.2 Estrategia de Persistencia

Estrategia	Descripción
IDs sin autoincremento	UUIDs generados en dominio (ValueGeneratedNever). Facilita creación en memoria.
Enums como string	HasConversion<string>() para legibilidad y compatibilidad.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

Cascade Delete	Sub-entidades eliminadas en cascada si se elimina el RegistryCase.
Retry Policy	EnableRetryOnFailure: 3 reintentos, delay máx. 5 segundos.
Command Timeout	3 minutos de timeout por comando SQL.

7.3 Repositorio: RegistryCaseRepository

Método	Descripción
AddAsync(case)	Persiste un nuevo RegistryCase.
GetByIdAsync(id)	Carga el caso completo con Include de las 4 colecciones.
GetByFilingNumberAsync(filingNumber)	Búsqueda exacta por número de radicado.
ExistsCorrespondenceOutputNumberAsync(number, excludedId)	Verifica unicidad del número de salida.
SearchAsync(query)	Búsqueda filtrada con paginación. Ordenado por FiledAt descendente.
SaveChangesAsync()	Persiste todos los cambios pendientes.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

8. Frontend Angular — Módulo de Registro

Diagrama de Componentes — Frontend Angular

RegistryWorkbenchPage	Componente principal OnPush + Signals
RegistryApiService	HTTP client + unwrap ResultModel<T>
Search Panel	7 filtros: radicado, fecha, asunto, destinatario, clasificación, dependencia, usuario
Results Table	12 columnas con paginación client-side y selección múltiple
Output Dialog Modal	Wizard 3 pasos: Nueva salida / Detalle / Acciones (decisión, docs, notif, ejecutoria, anotación)
Sync Service	syncAssignedCorrespondenceToRegistry() — fallback para cuando Kafka no llega

8.1 Estructura de Archivos

Archivo	Carpeta	Descripción
registry.models.ts	domain/	Tipos TypeScript: interfaces para RegistryCaseSummary, RegistryCaseDetail y todos los DTOs de request.
registry-api.service.ts	infrastructure/	Servicio Angular que encapsula todas las llamadas HTTP. Maneja unwrapping de ResultModel<T>.
registry-workbench.page.ts	presentation/ pages/	Componente principal con lógica de presentación. OnPush. Angular Signals.
registry-workbench.page.html	presentation/ pages/	Template con tabla, panel de filtros, panel de opciones y modal de detalle/creación.
registry-workbench.page.scss	presentation/ pages/	Estilos del componente.
registry.routes.ts	(raíz)	Rutas lazy del módulo de Registro.

8.2 RegistryApiService

Servicio singleton (providedIn: root). Utiliza HttpClientService del core y la URL base de APP_CONFIG.api.registry.

Método	Descripción

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

search(query)	Búsqueda con filtros y paginación. Retorna Observable<RegistryCaseSummary[]>.
getById(id)	Detalle completo. Retorna Observable<RegistryCaseDetail>.
create(request)	Crea un nuevo trámite.
assign(id, request)	Asigna responsable al trámite.
addDocument(id, request)	Adjunta documento o evidencia.
registerNotification(id, request)	Registra notificación al interesado.
resolveRegistryRequest(id, request)	Aplica decisión final.
startExecutoryPeriod(id, request)	Inicia periodo de ejecutoria.
markResourceFiled(id, request)	Registra si el interesado presentó recurso.
addBulkAnnotation(id, request)	Agrega anotación masiva.
registerCorrespondenceOutput(id, request)	Registra número de salida.
close(id, request)	Cierra el trámite.

El método privado unwrap<T>() maneja variaciones de respuesta: ResultModel<T> en camelCase o PascalCase, array directo o null.

8.3 RegistryWorkbenchPage — Estado Reactivo

Signals:

Signal	Tipo	Propósito
cases	RegistryCaseSummary[]	Lista de trámites cargados desde la API.
selectedCase	RegistryCaseDetail | null	Trámite seleccionado en el modal.
selectedIds	Set<string>	IDs seleccionados en la tabla (checkboxes).
loading	boolean	Indicador de carga.
saving	boolean	Indicador de guardado.
showOutputDialog	boolean	Visibilidad del modal.
creatingOutput	boolean	true = modal creación, false = edición.
page / pageSize	number	Paginación client-side.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

Computed signals: rows (filas mapeadas), visibleRows (slice paginado), totalPages, selectedCount.

8.4 Sincronización Frontend-Backend

En ngOnInit se invoca loadCases() que ejecuta dos pasos en secuencia:

- **1.** syncAssignedCorrespondenceToRegistry(): consulta hasta 3 endpoints de CorrespondenceManager en paralelo, filtra radicados de la dependencia Registro y crea los que no existan. Es un mecanismo de respaldo al worker de Kafka.
- **2.** registryApi.search(): carga hasta 200 trámites y actualiza el signal cases.

8.5 Flujo de Uso en la Interfaz

Acción	Descripción
Visualizar bandeja	Tabla de "Salidas de Registro" con filtros por radicado, fecha, asunto, destinatario, clasificación, dependencia y usuario.
Nueva salida	Botón "Nueva salida" abre modal en modo creación. Al guardar: crea RegistryCase y registra número de salida.
Ver/Editar trámite	Clic en radicado carga detalle completo en modal. Acciones: asignar, adjuntar docs, notificación, decisión, ejecutoria, cierre.
Exportar	Exportación a Excel o PDF. El PDF se abre en ventana nueva para impresión.
Prerrequisitos de cierre	El modal muestra indicadores TRD, Notificación, Salida, Recurso y mensaje "Falta:" con pendientes.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

9. Clasificaciones de Respuesta

El frontend define 7 clasificaciones de respuesta que determinan el tipo de proceso y el tipo de documento a generar. Son códigos internos usados como CategoryCode en el RegistryCase:

Código	Etiqueta	ProcessType	DocumentType	Decisión
RESP_REGISTRY_APPROVED	Respuesta - registro aprobado	RegistryRequest	RegistryCertificate	Registered
RESP_REGISTRY_RETURNED	Respuesta - devolución de solicitud	RegistryRequest	ReturnLetter	Returned
RESP_REGISTRY_REJECTED	Respuesta - negación de registro	RegistryRequest	RejectionResolution	Rejected
RESP_PETITION_RIGHT	Respuesta derecho de petición	PetitionRight	PetitionResponse	—
RESP_MODIFICATION	Respuesta a modificación	Modification	AdministrativeAct	—
RESP_CANCELLATION	Respuesta a cancelación	Cancellation	AdministrativeAct	—
RESP_REMISSION	Oficio remisorio	RegistryRequest	RemissionLetter	—

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno

10. Seguridad y Consideraciones Técnicas

10.1 Autenticación

Todos los endpoints requieren el header Authorization: Bearer <token>. La validación del token se delega a la infraestructura de autenticación del gateway o middleware configurado.

10.2 CORS

Se aplica la política DNDA-CORS. Si Cors:AllowedOrigins está configurado, solo esos orígenes tendrán acceso. En desarrollo con array vacío, se permite cualquier origen.

10.3 Manejo de Errores

UseExceptionHandler captura excepciones no controladas, registra con ILogger y devuelve HTTP 500. Los errores de negocio (400, 404, 409) son mapeados por el helper ToActionResult.

10.4 Resiliencia del Consumer Kafka

Patrón dead-letter: ante DbUpdateException o errores inesperados, confirma el offset del mensaje y continúa con el siguiente. El error queda registrado para revisión manual. JsonException también confirma el offset sin relanzar excepción.

10.5 Serialización JSON

Enums serializados como strings (JsonStringEnumConverter). PropertyNameCaseInsensitive=true para tolerancia de mayúsculas/minúsculas en requests y mensajes Kafka.

© 2026 Dirección Nacional de Derecho de Autor — Uso Interno