

Modelos

- [Modelos de Datos \(Eloquent\)](#)

Modelos de Datos (Eloquent)

Este documento condensa los conocimientos y entidades de la capa de datos (`app/Models`) del sistema **GNU**, describiendo la responsabilidad primordial de cada Modelo y sus relaciones principales.

? Arquitectura Institucional y Acceso

1. `Dependencia`

- Entidad que representa a las áreas operativas, oficinas o secretarías de la Institución.
- Se relaciona fuertemente con los Expedientes actuando como creador y como **poseedor (holder)**. Puede agrupar a múltiples usuarios y participar de los "pases paralelos" mediante tablas pivote.

2. `User`

- Representa al usuario físico/agente que opera el sistema. Utiliza características de `Spatie/Permissions` (Roles), autenticación de Jetstream, y se encuentra asignado en "caliente" a una dependencia (`dependenciaActiva`) en la que opera temporalmente.

? Núcleo Documental (Expedientes)

1. `Expediente`

- Es el corazón y objeto transaccional principal del sistema. Funciona como un agrupador legal de los sucesos.
- Tiene dos relaciones temporales críticas: `dependencia_id` (la originaria que lo creó) y `holder_id` (El tenedor actual en el flujo lógico, quien lo tiene en "posesión").

2. `TipoExpediente`

- Modelo de configuración y lectura. Define de forma maestra la taxonomía (Ej: Acción de Pago, Licitación, Expediente Estándar), sus colores de interfaz, e íconos.

3. `ExpedienteManifest`

- Entidad puramente técnica. Modela la cadena de bloques (Blockchain interno) del expediente. Guarda instantáneas versionadas (`version`, `manifest_hash`, `prev_manifest_hash`, `state_hash`) garantizando integridad legal de manera inmutable tras cada movimiento.

4. `DependenciaExpediente`

- Modelo que actúa netamente como *Clase Pivote* (`Pivot`) con atributos adicionales (`activo`, `expira`) para sostener de forma estructurada el **compartir simultáneos paralelo** los expedientes.

? Solicitudes

1. `Solicitud`
 - Predecesora del Expediente. Es el formulario interactivo provisto en la mesa de entrada o self-service. Contiene los datos crudos provistos por el "Causante". Una vez aprobada se convierte en Expediente Oficial mediante la rutina de `caratular`.

?Comunicaciones

1. `Ccoo`
 - Modela una "*Comunicación Oficial*". Es una entidad de envío de oficios, resoluciones o mensajes entre una `dependencia_origen` y una `dependencia_destino`.

? Archivos y Adjuntos (Ecosistema Múltiple)

El sistema segrega fuertemente los subidos acorde al ámbito donde se agregaron:

1. `Archive`
 - Archivos y fojas PDF vinculadas directamente a un **Expediente** Oficializado. Está versionado (`hash`), auditado, y permite adjuntar `firmas`.
2. `Archivot`
 - Archivos transitorios vinculados inicialmente a una **Solicitud** que aún no es Expediente.
3. `ArchivoCcoo`
 - Disposición de almacenamiento enfocada a las Comunicaciones Oficiales (`Ccoo`).
4. `Firma`
 - Entidad complementaria al `Archive`. Captura los metadatos específicos leídos en un PDF (Nombre del certificado, fecha de validación legal del PKI, validez criptográfica) en un registro verificable.
5. (Nota: `File` es una clase obsoleta o fallida marcada en el código para deprendimiento futuro)

? Herramientas de Sistema

1. **Log**

- Modela la traza silenciosa de auditoría interna. Almacena la Acción, qué Usuario/Dependencia interactuó, sobre qué entidad (Solicitud/Expediente) y anexa factores forenses básicos de HTTP (IP, User Agent).

2. **Email**

- Réplica de correos institucionales inyectados. Guarda destinatario, remitente y el cuerpo limpio que luego se utiliza interactivamente con el controlador de Inteligencia Artificial de las bandejas integradas.