**Análisis general del modelo ER**

**Dominio y entidades (qué tablas habrá y para qué)**

1. **Usuarios**
   * **User**: persona que usa la plataforma. Rol: *artista*, *curador*, *coleccionista* (y opcional *admin*). Necesario para todas las acciones (subir NFT, revisar, ofertar).
   * **Wallet**: “billetera” por usuario, con **saldo disponible** y **saldo reservado** (se reserva al hacer ofertas exitosas). El enunciado exige validar saldo antes de aceptar una oferta y reservar fondos al registrar la oferta.
2. **Obras / NFTs**
   * **NFT**: obra con metadatos (nombre, descripción, tipo de archivo), **hash único** para identidad/autenticidad, **estado** de ciclo de vida (*pendiente/aprobado/rechazado/finalizado*) y referencias a **artista** y **propietario actual**. Al subir, se generan hash y correo de confirmación; pasa a “pendiente” mientras se valida.
3. **Curaduría / Validación**
   * **CurationReview**: revisiones realizadas por **curadores** con decisión (*aprobar/rechazar*), comentarios y tiempos. Esto alimenta el reporte “Eficiencia de curadores” (volumen, tiempos, tasas).
4. **Subastas**
   * **Auction**: se **crea automáticamente** cuando un NFT pasa a aprobado; **duración estándar 72h**, **precio inicial** (sugerido por artista si existe o **precio base** de la plataforma), y estado (*activa/finalizada*). También se notifica por correo y aparece en la galería de subastas activas. Es configurable (admin puede cambiar duración y precio base).
   * **(Decisión de diseño clave)**: ¿1 NFT = 1 subasta o permitir múltiples subastas históricas? El enunciado habla de “crear una subasta” al aprobar; si quieres re-subastar no vendidas, podríamos pasar a **1:N (NFT→Subastas)** con un campo “motivo/oleada”. Si lo mantienes simple para la rúbrica, **1:1** funciona y cumple.
5. **Ofertas**
   * **Bid**: oferta con **monto**, **momento exacto**, **postor** y **subasta**. Debe pasar validaciones: saldo suficiente, subasta activa/no finalizada, y que la oferta sea **mayor** a la actual (el ejemplo narra también una regla de incremento mínimo del 5%, pero el requisito formal pide al menos que sea mayor; si quieres, ese incremento puede quedar como **parámetro** de configuración). Se debe **registrar historial** y enviar confirmación por correo.
6. **Reservas y contabilidad interna**
   * **FundsReservation**: registra **reservas** de fondos por oferta aceptada (activo/liberado/aplicado). Esto permite reembolsar a perdedores o aplicar al ganador al finalizar. Está implícito en “reservar temporalmente el saldo” y “reembolsar perdedores”.
   * **Ledger** (opcional pero recomendable): **asiento contable** por débito/crédito (compra del ganador; pago al artista). Facilita auditoría y reportes financieros.
7. **Notificaciones**
   * **EmailOutbox**: bandeja de salida para correos del sistema (recepción de NFT, activación de subasta, confirmaciones de oferta, finalización a ganador/artista/perdedores). La finalización requiere expresamente esos correos.
8. **Parámetros / Configuración**
   * **Settings**: claves como **duración estándar (72h)**, **precio base (p. ej. 0.1 ETH)** y (si decides) **% mínimo de incremento**. El enunciado pide explícitamente que duración y precio base sean configurables por el administrador.
9. **(Para reportes)**
   * Con lo anterior, salen 4 vistas/consultas:  
     **Eficiencia curadores**, **Actividad de coleccionistas**, **Valorización de artistas en el tiempo**, **Métricas por período**. Los campos (tiempos, conteos, montos, tasas) están descritos en el enunciado.

**Alcance**: todo en **capa BD** (T-SQL, transacciones, SPs, triggers), sin app.

**Relaciones (cardinalidades en “modo hablado”)**

* **User 1—1 Wallet** (una billetera por usuario).
* **User (artista) 1—N NFT** (un artista puede crear muchas obras).
* **User (propietario) 1—N NFT** (propiedad se transfiere al ganador al finalizar).
* **NFT 1—N CurationReview** (pueden existir varias revisiones; sólo la última decisión válida cambia el estado).
* **NFT 1—1 Auction** (simplificado) **o** **NFT 1—N Auction** (si permites re-subastas).
* **Auction 1—N Bid** (múltiples ofertas por subasta).
* **User (coleccionista) 1—N Bid** (un usuario hace muchas ofertas).
* **User 1—N FundsReservation** y **Auction 1—N FundsReservation** (reservas ligadas a postor y subasta).
* **Auction 1—N Ledger**, **User 1—N Ledger** (asientos de débito/crédito en finalización).
* **EmailOutbox** se relaciona lógicamente con eventos (subida, aprobación, oferta, finalización).

**Ciclos de vida y reglas (para estados y procesos)**

1. **Creación/validación de NFT**
   * *Subir* → **PENDIENTE** (hash + correo) → *Curador aprueba/rechaza* → **APROBADO** (si aprueba) **o** **RECHAZADO** (si no). Aprobado **dispara subasta**.
2. **Subasta automática**
   * Al pasar a **APROBADO**: crear subasta (72h, precio inicial sugerido o base), estado **ACTIVA**, notificar artista y coleccionistas, mostrar en galería.
3. **Ofertas**
   * Validar: **saldo suficiente**, **subasta activa y vigente**, **monto mayor** a precio actual (si decides, aplicar **incremento mínimo configurable**). Registrar **historial**, **reservar fondos** y **confirmar por correo**.
4. **Finalización (batch cada ~5 minutos)**
   * Detectar vencidas, para cada subasta: **ganador = mayor oferta** (primera en empate), **transferir NFT** al ganador, **pagar artista**, **reembolsar perdedores**, **marcar finalizada**, **enviar correos** (ganador/artista/perdedores).

**Claves, restricciones y trazabilidad (qué cuidar en el diseño)**

* **Identidad única de obra**: hash de contenido **único** para detectar duplicados.
* **Estados y fechas coherentes**: subasta end > start; una subasta sólo puede estar **ACTIVA** si el NFT está **APROBADO**.
* **Integridad financiera**: saldo ≥ 0, reservado ≥ 0; **oferta** sólo si hay **saldo disponible** suficiente; reservas pasan a **APLICADO** (ganador) o **LIBERADO** (perdedores).
* **Historial público de ofertas** y **auditoría** (reservas/ledger) para soportar reportes.
* **Configuración flexible**: **duración estándar** y **precio base** administrables (parámetros).
* **Alcance técnico**: todo se implementa con **T-SQL** (SPs, triggers, transacciones, funciones), **sin app**.

**Decisiones abiertas (valen la pena acordarlas antes del DDL)**

* **Una o varias subastas por NFT** (re-subastar no vendidas). Recomendación: empezar con **1:1** y dejar **1:N** como extensión.
* **Regla de incremento mínimo**: el ejemplo usa **5%**; el requisito formal exige “mayor que la actual”. Podemos dejar un **parámetro opcional de incremento mínimo** para alinearlo con la narrativa.
* **Comisiones de la plataforma**: no se mencionan; si decides agregar, impacta **Ledger** y **liquidación**.
* **Múltiples roles por usuario** (ej. un curador también colecciona): si lo permites, modela **UserRole** (N:M) en lugar de un único rol.

Si te parece bien este mapa conceptual, pasamos a dibujar el **ER a nivel lógico** (nombres de entidades/relaciones y cardinalidades) y luego a tu **diccionario de datos** (atributos, tipos, dominios y reglas), que además la rúbrica pide explícitamente.