

Diseño - Proyecto1 - DPOO

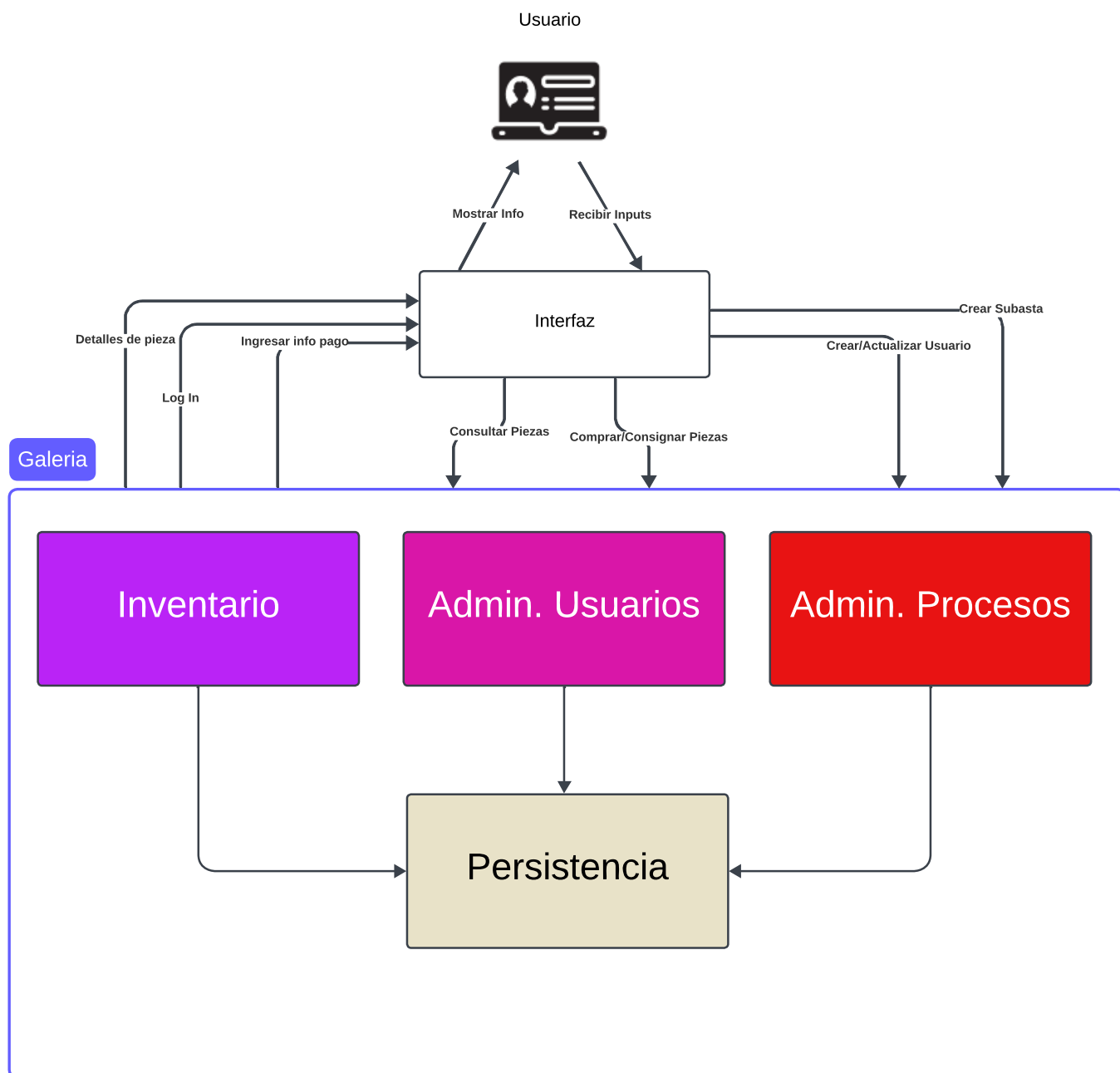
Andres Tello, Emmanuel Blanco y Elizabeth Labarces

202223807, 202312743 y 202313672

Abril 2024

1 Contexto

Antes de empezar el proceso de diseño definimos las funcionalidades fundamentales para el programa (mostradas en la figura 1). Esto tiene el propósito de facilitar el desarrollo del sistema y la interfaz, ya que sabemos que cosas debe retornar y pedir el sistema y la interfaz. Además, al hacer este proceso podemos desarrollar la interfaz como deseemos, puesto que al diseñar el sistema haremos métodos con los cuales conseguir la información detallada en el proceso anterior.



2 Nivel 1

Estereotipos

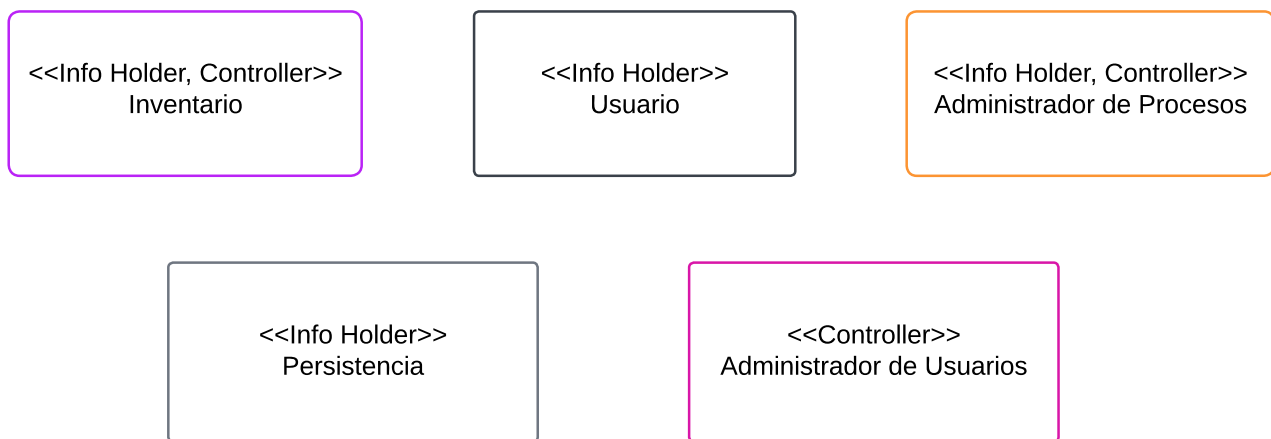


Figure 2: Estereotipos

3 Componentes Candidatos y Estereotipos

El sistema que se diseñó se divide en tres secciones grandes: **el inventario, los procesos y los usuarios**. Cada uno de estos tiene varios roles y varias responsabilidades. Cada una de estas secciones será un **information holder**, puesto que cada una tiene datos que cargar. Además, cada una tendrá un **controller** que recibe las entradas del usuario para esa sección e invoca los métodos apropiados para esa sección. Veamos cada sección con más detalle.

3.1 Usuarios - Information Holder y Controller

La sección de usuarios es donde se guarda toda la información de los usuarios, aun así si son empleados administrador es o clientes. Como esta sección esta dedicada a guardar información y verificar quien esta accediendo a la aplicación, consideramos que Usuarios debe tener un information holder y un controller. Con eso en mente, estas son las responsabilidades de Usuarios

- Login y Logout
- Búsqueda de usuarios
- Cargar y guardar usuarios (Persistencia)
- Cambiar información de usuario

3.2 Inventario - Information Holder, Controller y Coordinator

El inventario es donde se guardará toda la información de las piezas sin importar su estado (en exhibición, en bodega, en devolución o en el historial). Es por esto que naturalmente debe ser un **information holder**. Dicho eso, hay responsabilidades que corresponden a un **controller**, como mover piezas de un estado a otro o buscar piezas en particular. Las responsabilidades que tiene el inventario son las siguientes:

- Buscar Piezas
- Mover Piezas (de estados)
- Registrar Pieza
- Disponible (revisa que la consignación de la pieza siga valida)

3.3 Procesos - Information Holder, Service Provider, Controller y Coordinator

La sección de procesos se encarga de todas las ventas, subastas y consignaciones. Como las otras secciones, esta debe guardar la información de las subastas abiertas, por lo que es un **Information Holder**. Sin embargo, esta encargado de hacer las ventas, subastas y consignaciones, que lo hacen un **Service Provider**. También debe manejar la confirmación de ventas y subastas y la creación de nuevas subastas por lo que también es **Controller**. Finalmente, también debe tener capacidades de **Coordinator** ya que las subastas tienen un tiempo limite, por lo que debe haber algún componente independiente que revise las fechas limite de las subastas y las cierre cuando sea el momento.

Estas son las responsabilidades del administrador de procesos:

- Registrar compra

- Bloquear o desbloquear piezas
- Generar recibo
- Aumentar oferta subasta
- Iniciar y terminar subasta
- Actualizar subastas abiertas
- Guardar subastas abiertas

3.4 Conclusiones de Componentes

Cada una de las secciones que reconocimos tiene un controlador, por lo que a la sección de procesos se les agregaron objetos llamados Administrador de procesos y Administrador de usuarios respectivamente, para la sección del inventario dejamos el objeto inventario y le agregamos las funciones para poder manejar toda la información.

3.5 Colaboraciones

Aquí se detallará cuales son las colaboraciones y como suceden en la galería.

3.5.1 Ventas y Subastas

Para las ventas que hará el sistema el usuario buscará la pieza que desee comprar y optará por comprarla. Una vez haya sucedido esto el sistema deberá hacer la siguiente operación:

Cuando se dice que el controlador "busca" algún elemento se refiere a que ejecuta una operación de búsqueda dentro de los controladores de cada sección.

1. El Usuario entra a la aplicación y hace LogIn con las credenciales guardadas en Administrador de Usuarios
2. Los usuarios ingresan al inventario y eligen una pieza que quieran comprar
3. Se inicia el proceso de venta en el Administrador de Procesos
4. El Administrador de Procesos verifica que el usuario tenga el monto suficiente para comprar la pieza y guarda su información
5. Se guarda la compra en el historial del usuario en el Administrador de Usuarios y se guarda la venta de la pieza en el Administrador de Procesos

Proceso Venta

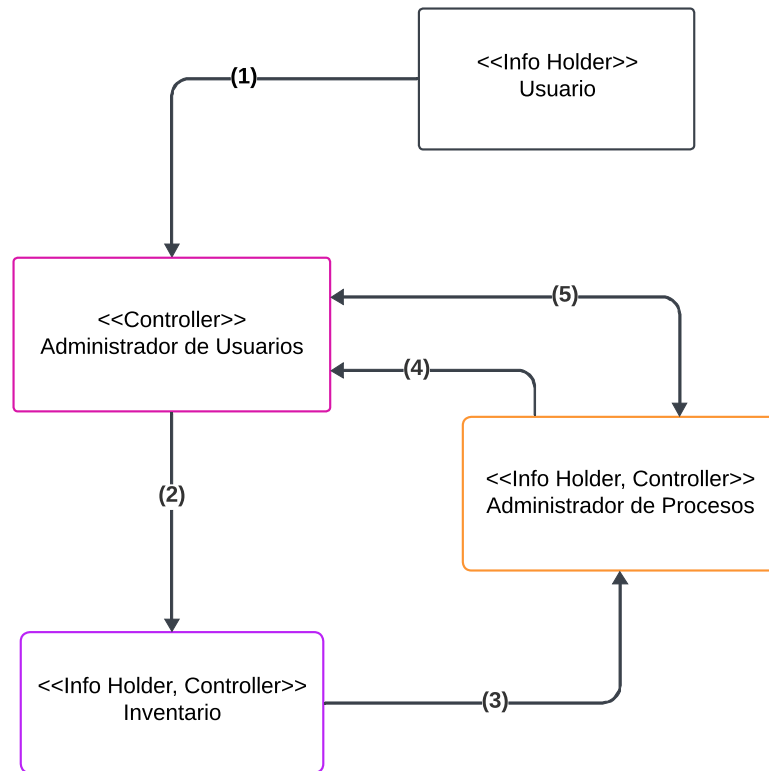


Figure 3: Diagrama para Venta

Para registrar una nueva oferta en una subasta, el usuario nuevamente buscará la subasta y optará por hacer una oferta e ingresará la cantidad de plata a la que quiere incrementar. En el siguiente diagrama mostramos como sucederá esta operación.

1. El cliente Hace el LogIn
2. El Usuario accede al inventario y elige una pieza que esta en subasta,

donde decide ingresar una oferta

3. El administrador de procesos comienza la nueva oferta y le pide a que precio quiere hacer su oferta.
4. El administrador de usuarios revisa que tenga suficiente monto para hacer la oferta.
5. Una vez se haya confirmado el monto se actualiza la información en el administrador de procesos

Proceso Nueva Oferta

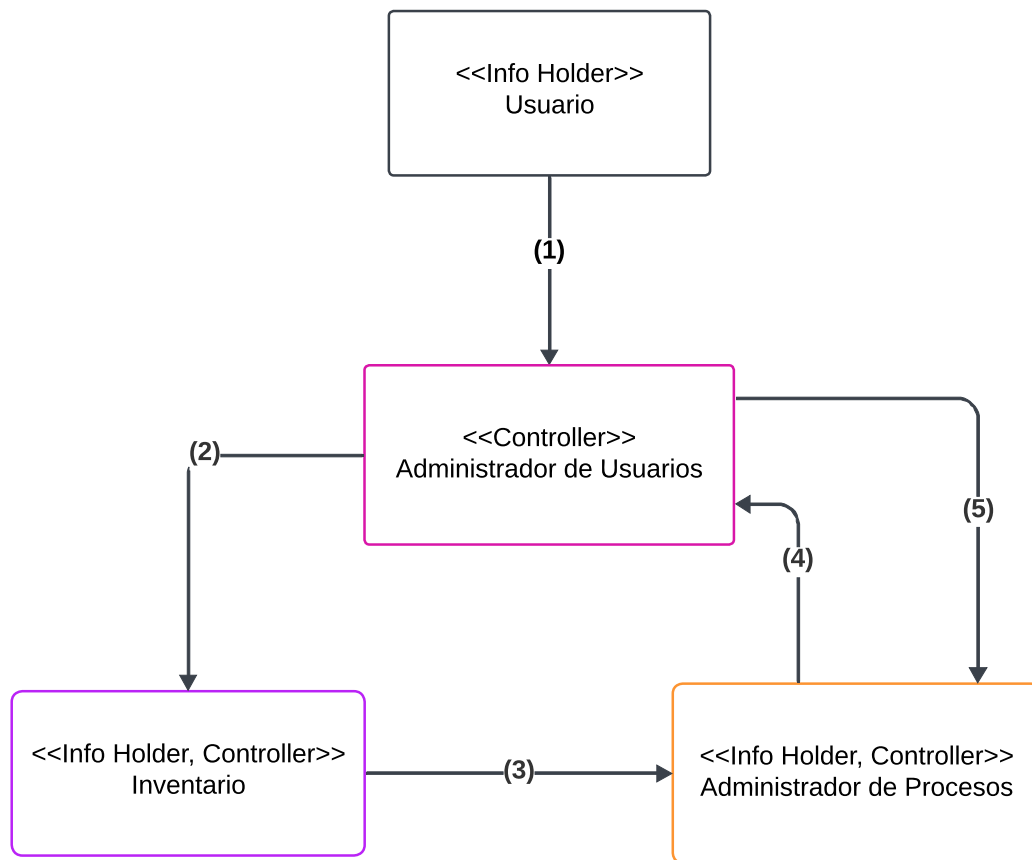


Figure 4: Diagrama para Nuevas Ofertas

3.5.2 Devolución

Si una pieza termina su consignación y no se ha vendido o subastada la pieza debe ser devuelta al cliente que la consigno. Para poder devolverla necesitamos la información de contacto del propietario. Esta información la podemos conseguir buscando el usuario en el Administrador de Usuarios

La devolución se verá de la siguiente manera:

1. El Inventario encuentra una pieza con la consignación vencida
2. Busca el propietario en el administrador de usuarios
3. Se actualiza el estado de la pieza en el inventario, queda en devolución.

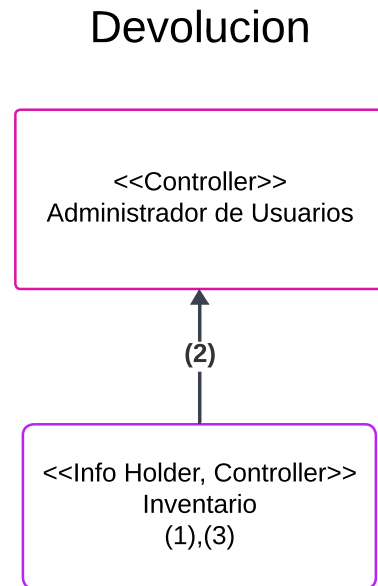


Figure 5: Diagrama para Devoluciones

3.5.3 Verificar un Proceso

Mientras que piden que los administradores puedan confirmar procesos, en el sistema se genera la venta una vez que este ya está confirmado, debido a que la

venta funciona como un recibo o un registro en el sistema.

3.5.4 Consignar una Pieza

Para una consignación requerimos que el cliente nos de su información de contacto, para eso lo sacamos del Administrador de Usuarios y guardamos esa información en una nueva instancia de pieza y agregamos eso a consignaciones.

1. El cliente hace LogIn con las credenciales en el administrador de usuarios
2. El cliente decide hacer una consignación y comienza el proceso en el administrador de procesos
3. El administrador de procesos le pide la información del usuario al administrador de usuarios
4. El inventario guarda la nueva pieza en la bodega

Consignación

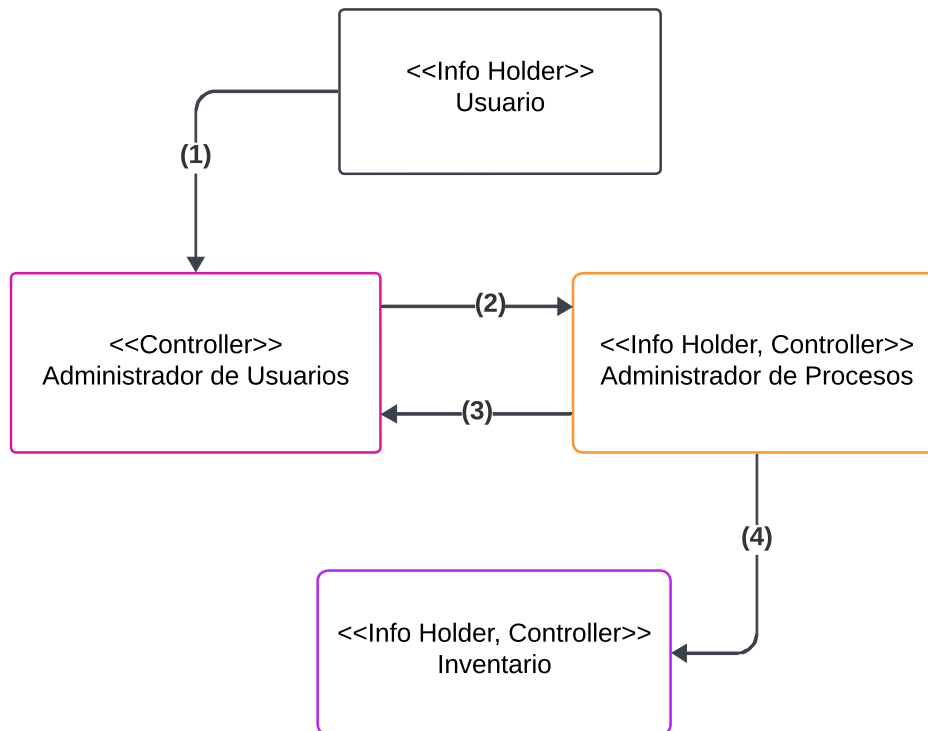


Figure 6: Diagrama para Consignación

3.6 Diagrama Final Nivel 1

Teniendo en cuenta las responsabilidades de cada sección y los elementos que necesiten para cumplirlas llegamos al siguiente diseño:

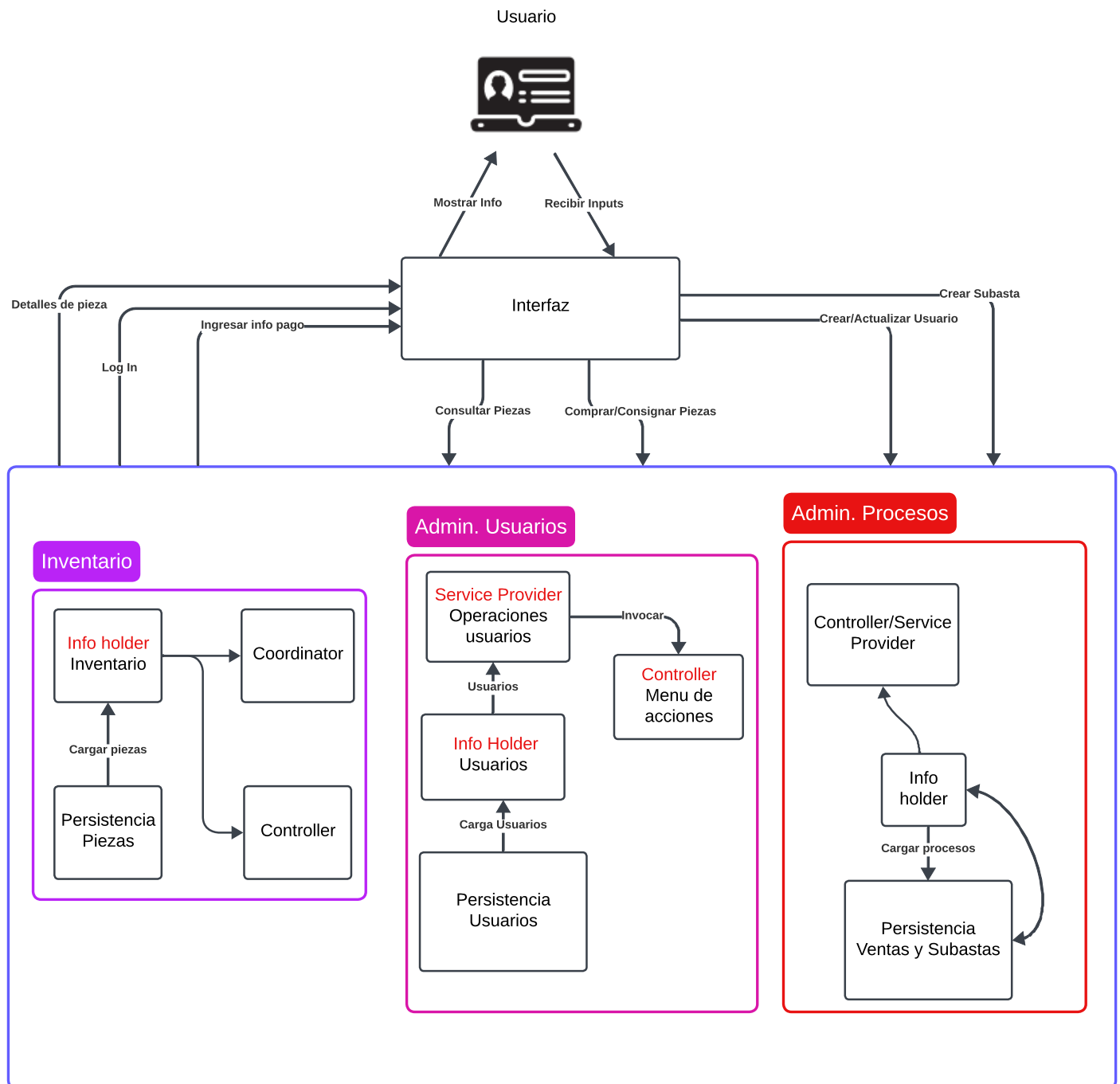


Figure 7: Componentes básicos del sistema de la galería

4 Nivel 2

Tras plantear la galería como lo mostramos antes hacemos la siguiente división de los componentes.

5 Componentes Candidatos y estereotipos

5.1 usuarios.modelo

La sección Usuario esta encargada de guardar y manejar toda la información de los usuarios, sean clientes, empleados o administradores.

Esta sección debe cumplir con las siguientes responsabilidades:

- Crear usuarios nuevos
- Actualizar usuarios
- Buscar usuarios
- Reconocer Usuarios
- Reconocer el tipo de usuario
- Log-In
- Log-Out
- Cargar usuarios

En el nivel anterior se menciona como esta sección claramente debe cumplir las funciones de information holder, además de poder interactuar con la persona detrás de la computadora. Es por eso que anteriormente habíamos mencionado la necesidad de que tuviera un controller.

Con esto en mente distribuimos la sección de la siguiente manera:

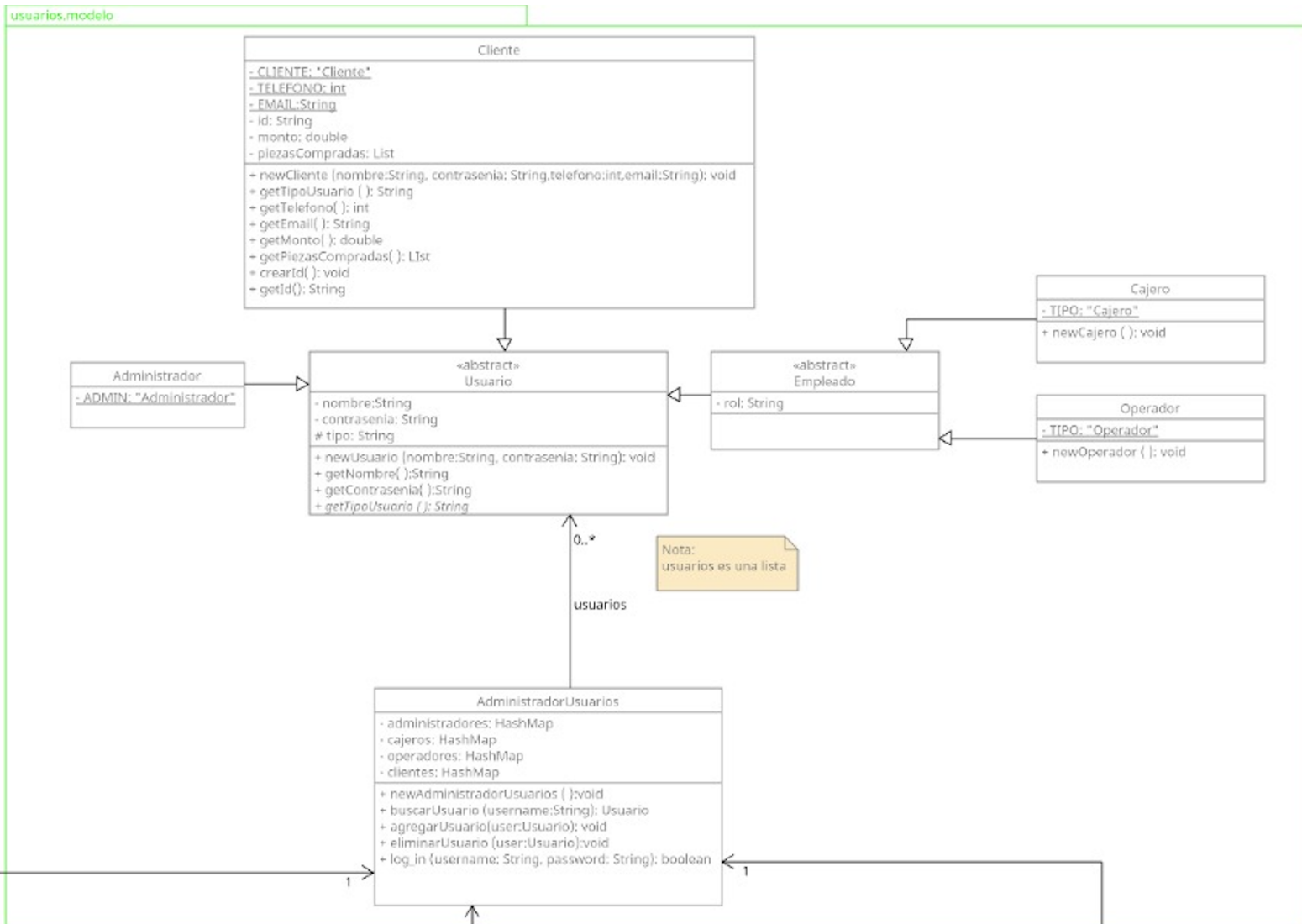


Figure 8: Usuarios Nivel 2

La sección usuarios es un paquete que contiene las siguientes cosas:

- Administrador de usuarios
 - Tiene todas las funciones que manejan la información de los usuarios
- Usuario
 - Una clase abstracta que funciona como la estructura para todos los usuarios

- Cliente
 - Una clase que hereda de Usuario, esta clase contiene toda la información de un cliente de la galería
- Administrador
 - Es una clase que diferencia a estos usuarios de otros empleados
- Empleado
 - Es una clase abstracta que guarda el rol del empleado
- Cajero y Operador
 - Son clases que heredan de Empleado y sirven para diferenciar los diferentes roles que pueden tener los empleados

5.2 inventario.modelo

La sección inventario.modelo es la misma que la sección Inventario en el Nivel 1, queríamos mantener el nombre para la clase principal de la sección por lo que cambiamos el nombre de la sección a inventario.modelo.

Esta sección está encargada de manejar y guardar toda la información de las piezas. Hay una variedad de tipos de piezas que puede recibir la galería (pinturas, esculturas, escrituras, vídeos, colecciones, etc) y estas pueden estar en uno de varios estados (exhibición, bodega, devolución, venta y subasta) por lo que es importante poder distribuirlas y separarlas de una manera organizada.

Esta sección debe tener la capacidad de ejecutar las siguientes operaciones:

- Buscar una pieza
- Cambiar el estado de una pieza
- Retirar pieza del inventario
- Revisar disponibilidad
- Registrar una nueva pieza
- Cargar y guardar datos de las piezas

En el nivel 1 mencionamos que esta sección tendría responsabilidades de un information holder, un controller y un coordinator. Estas responsabilidades ahora se distribuyen de la siguiente manera;

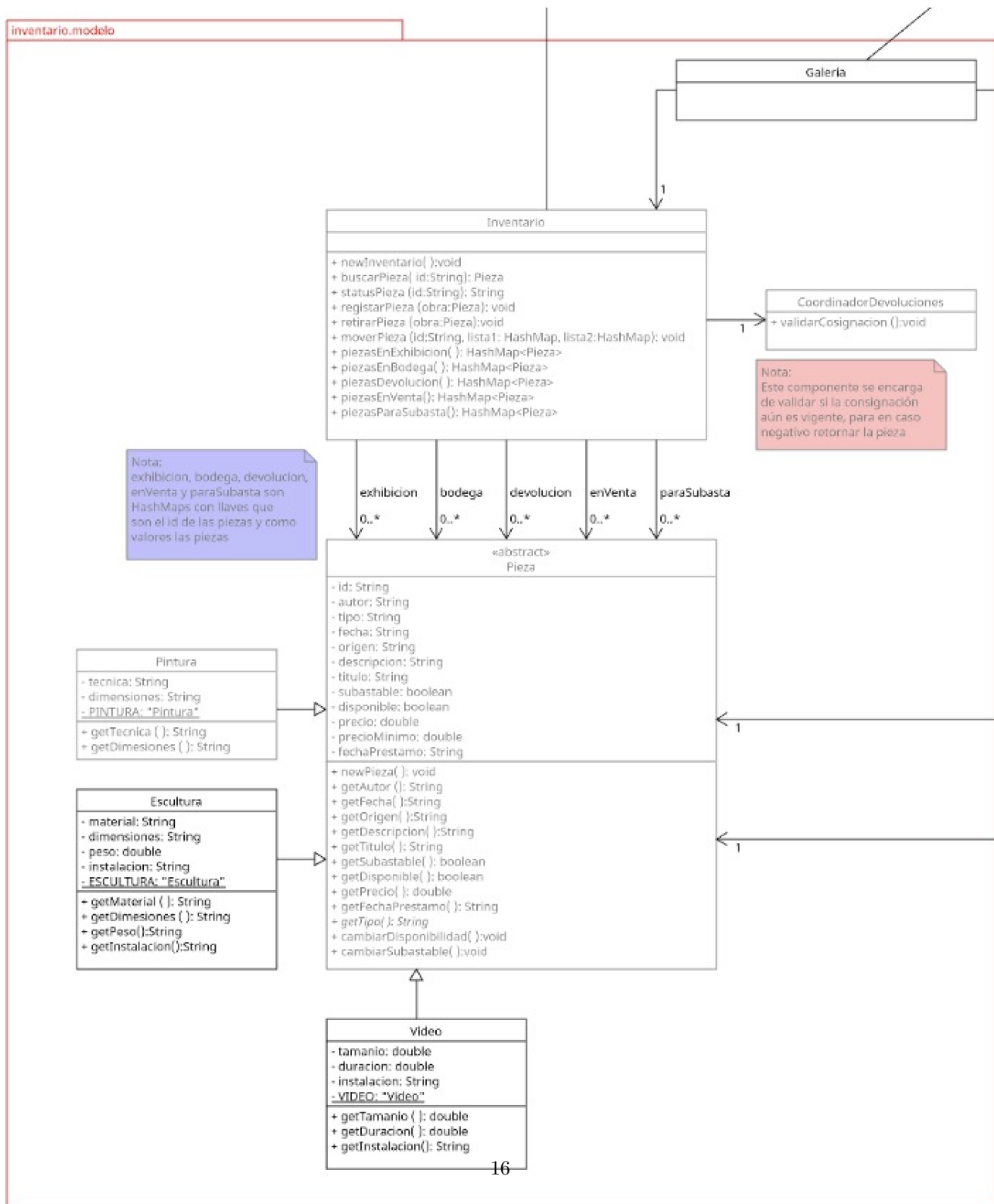


Figure 9: Storage Nivel 2

- Inventario es un Controlador, tiene los métodos para buscar piezas y la información de ellas.
- CoordinadorDevoluciones es un coordinador, se encarga de revisar que las consignaciones de las piezas sean validas y cuando no lo son les cambia el estado a devolución
- Pieza es una clase estructural, le da atributos y métodos esenciales a todas las piezas
- Pintura, Escultura y Video son information holders, son los objetos que van a contener la información de las piezas que tenga la galería

Para la persistencia de inventario.modelo se desarrollo el siguiente proceso, en el cual mostramos la carga de datos de inventario.

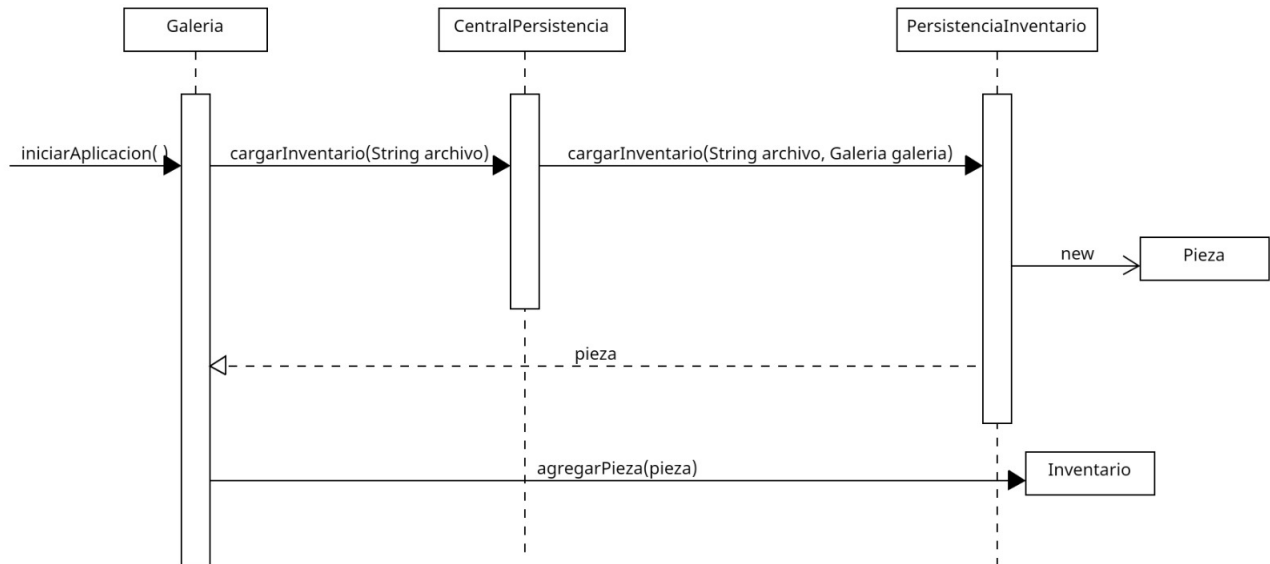


Figure 10: Diagrama de Secuencia Persistencia

5.3 procesos.modelo

La sección de procesos anteriormente tenía los deberes de un information holder, un service provider, un controller y un coordinator. Además debe cumplir con las siguientes responsabilidades:

- Hacer una compra
- Bloquear compras o subastas

- Crear un recibo para cada compra y subasta
- Retirar pieza
- Aumentar oferta de subasta
- registrar una venta o subasta nueva

Para cumplir con estas responsabilidades dividimos la sección en las siguientes clases:

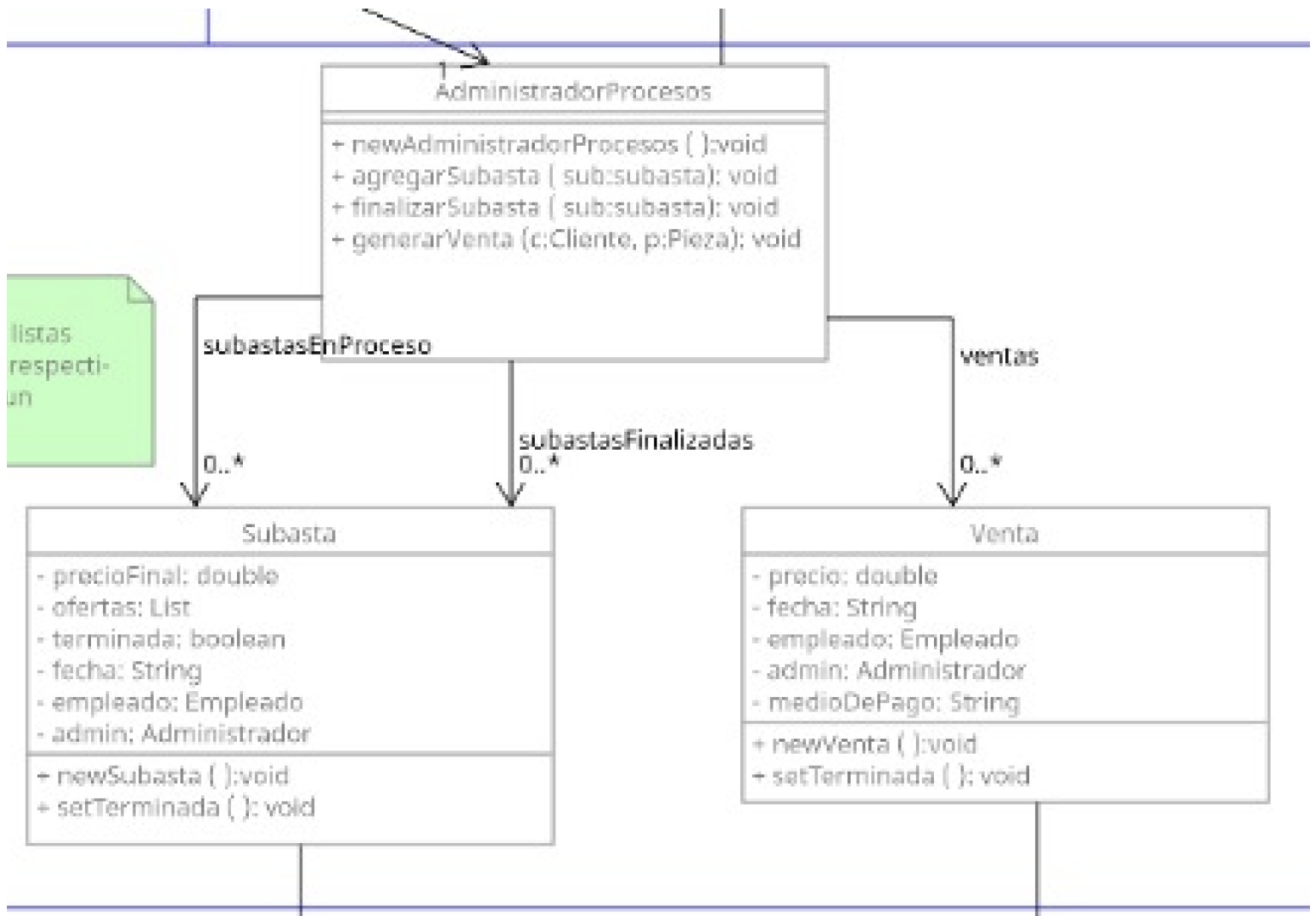


Figure 11: Procesos Nivel 2

Nota: En la Imagen se ve un elemento aparte de las clases mencionadas. Esto es una nota que se agrego para aclaraciones, se puede ver en el diagrama final

- El administrador de procesos es un controller y se encarga de todos los metodos de busquedas de información y manejo de los datos
- Venta es un information holder que tiene toda la información de ventas ya procesadas
- Subasta es otro information holder que contiene toda la información acerca de una subasta, sea una que está activa o una que ya termino.

5.4 Colaboraciones

Frente a las colaboraciones, la galería manejará una estilo centralizado, donde el controlador buscara toda la información que necesite delegando trabajos a los controladores de cada sección.

5.5 Diseño Final

Para resumir el proceso, se hizo un sistema de manejo de datos de una galería con un estilo centralizado. Como cada sección del sistema trata con información diferente, optamos por darle un controlador y un information holder a cada una y agregamos un coordinator en el caso donde el sistema tenga que actualizar información por si solo. Además agregamos conexiones entre los controladores y el controlador central (galería) para las colaboraciones independientes.

Así es como nos quedo el diagrama final: *Nota: Para una imagen con mejor resolución se pueden observar los diagramas UML finales en la carpeta "UML" en el repositorio de la entrega* Y este es un diagrama de clases de alto nivel:

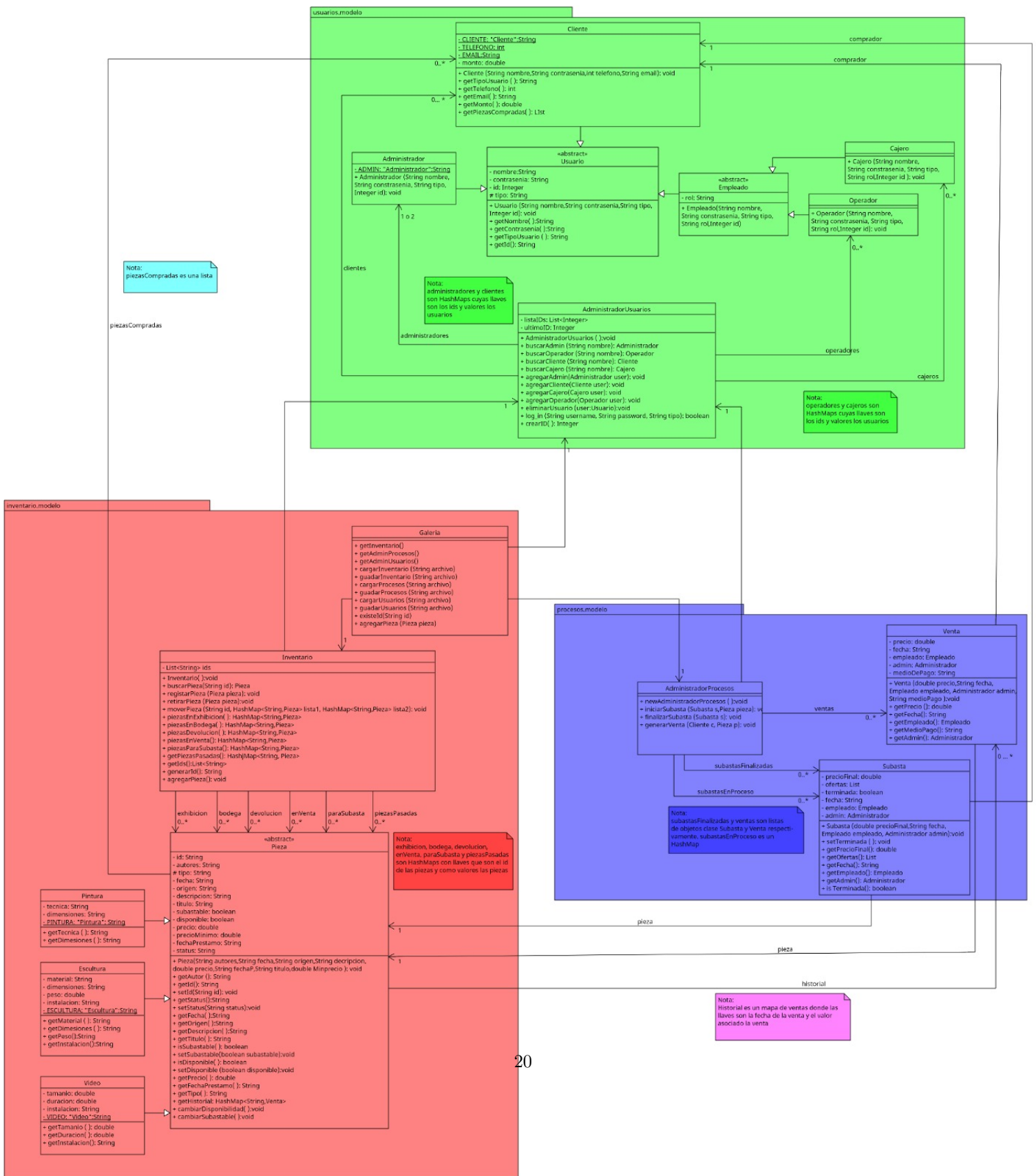


Figure 12: Diseño Final Completo

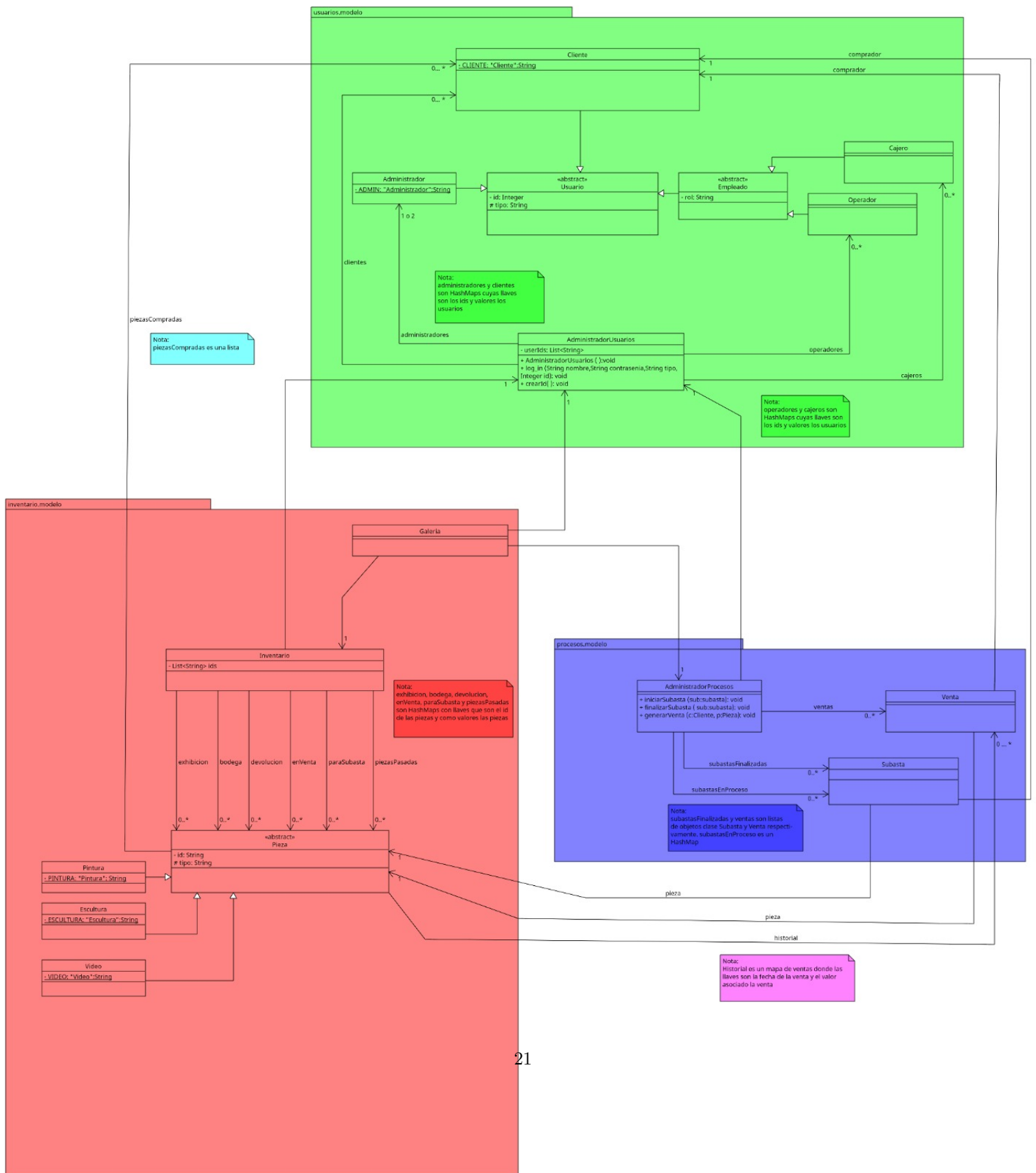


Figure 13: Diseño Final Sin Métodos