UNComa

Trabajo Final de Programación Orientada a Objetos

Alumno: Iovaldi, Sebastián Ángel

Legajo: FAI-3482

Año de Cursada: 2022

Tabla de contenido

Diagrama del Dominio

Se necesita desarrollar un sistema gestor de préstamos de libros a bibliotecas. El mismo tiene una colección de bibliotecas con las que trabaja y debe poder registrar usuarios que podrán pedir los libros que deseen online a las bibliotecas existentes en el sistema.
Las bibliotecas ofrecen para préstamo libros impresos, libros electrónicos y audiolibros.

Al momento de crear usuarios, se necesita registrar nombre, apellido, dni y contraseña. Luego, se debe dar la opción de agregar sus bibliotecas de interés, es decir aquellas en las que podrá solicitar préstamos.

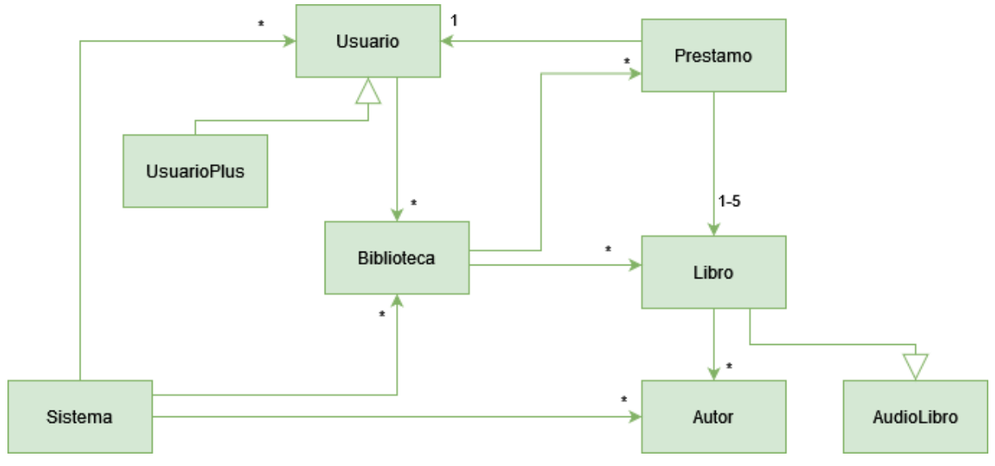
De cada biblioteca (identificador, dirección, nombre, teléfono de contacto y nombre del representante) es necesario saber que libros posee. A su vez, de cada libro se necesita su nombre, editorial, año de edición y sus autores. Un libro puede tener 1 o más autores.

Un usuario puede solicitar un préstamo de libros (entre 1 y 5) en una misma biblioteca. Es muy frecuente que los socios se olviden de entregar los libros a tiempo, por lo que el sistema debe tener la capacidad de encontrar qué libros están en deuda y que usuarios los tienen en su poder.
El sistema multa con una suspensión de ***k dias*** al usuario que se retrasa en la devolución de los libros, es decir que por esa cantidad de días no podrá solicitar nuevos préstamos.

Un usuario puede solicitar préstamos de libros impresos y libros electrónicos, pero los audiolibros solo están disponibles para un tipo particular de usuario: *el usuario plus*. Un **usuario plus** es el que paga una cuota mensual al sistema para tener acceso a todo tipo de material disponible en las bibliotecas.

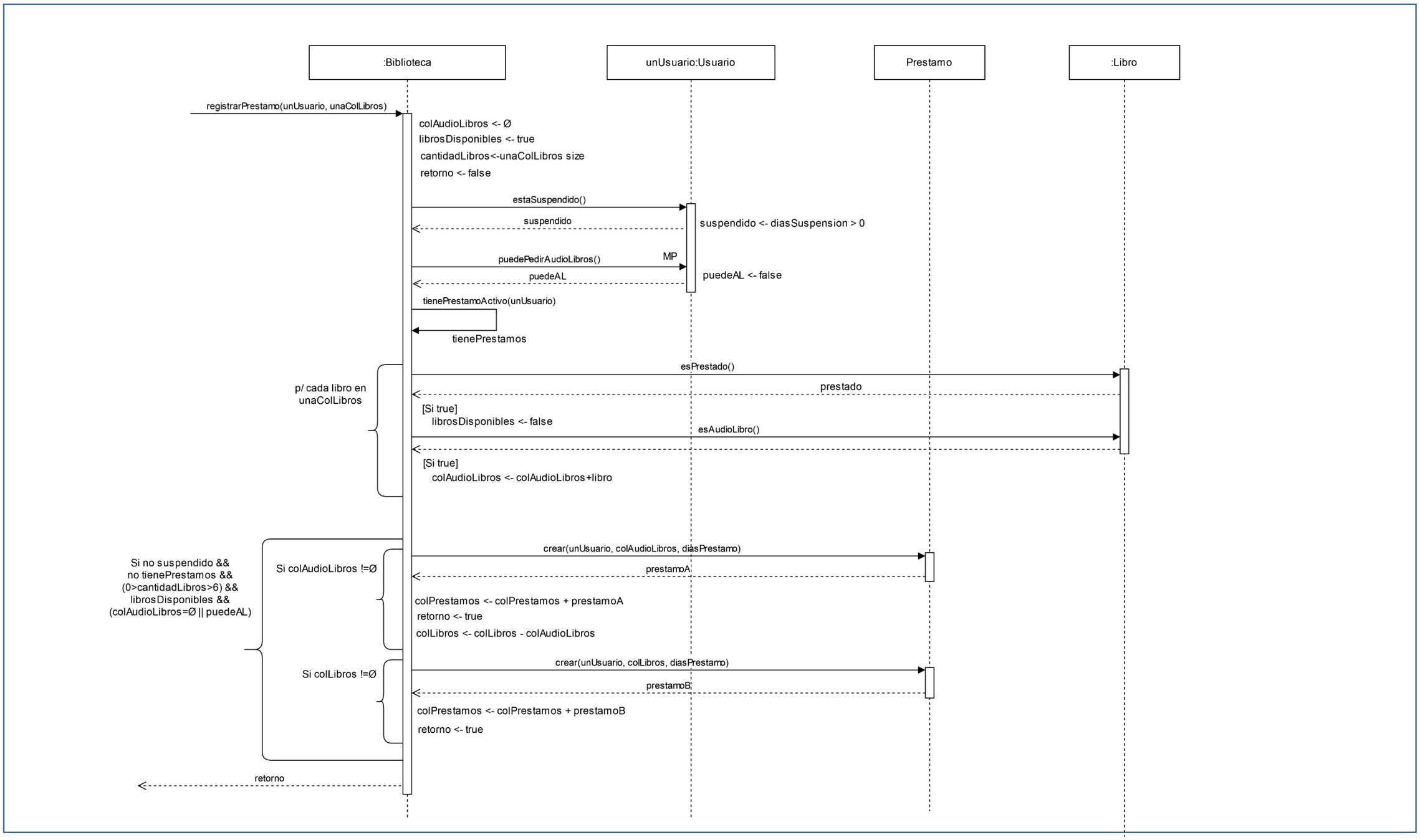
Cada biblioteca define por cuánto tiempo presta sus libros impresos y electrónicos. Los audiolibros, por otro lado, se prestan por la mitad del tiempo y solo a los usuarios plus del sistema.

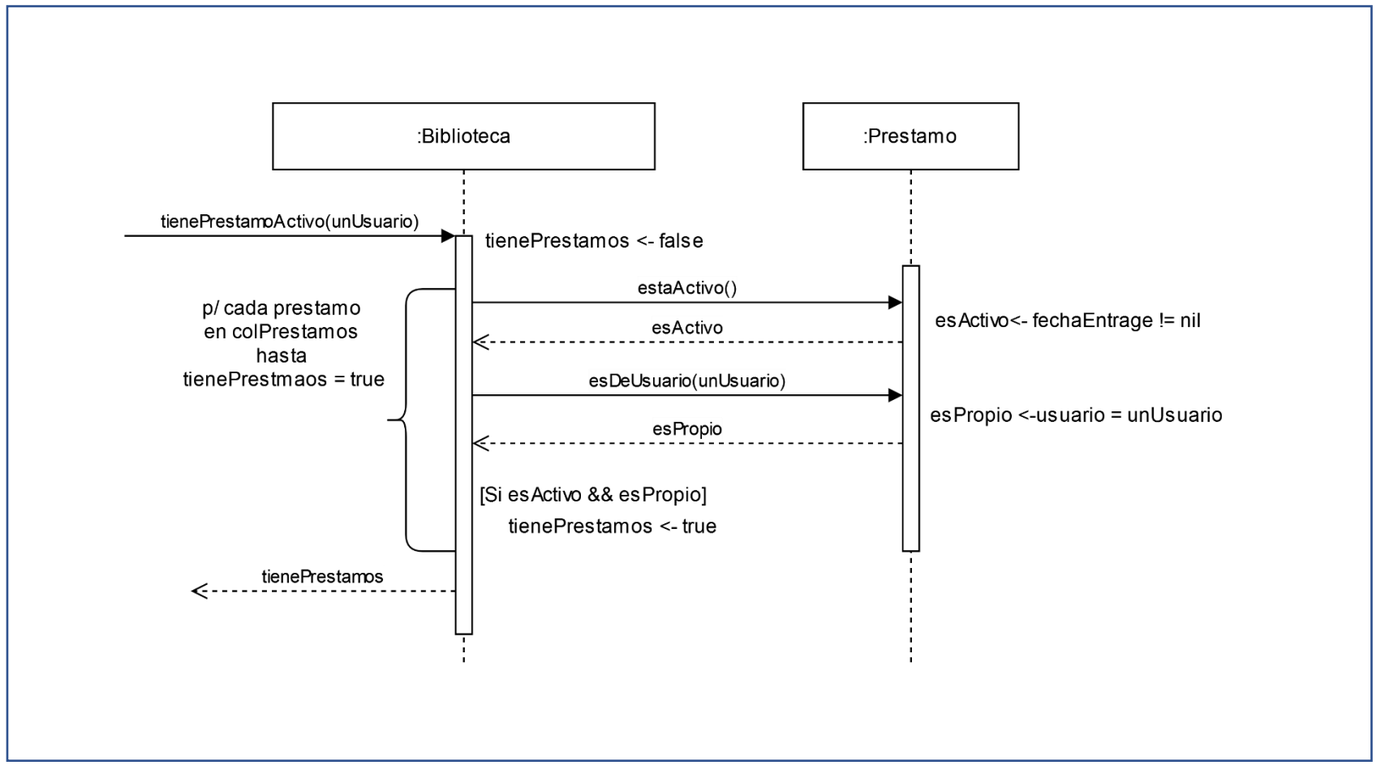
Para evitar problemas, cada biblioteca puede hacer 1 solo préstamo de libros a un mismo usuario a la vez. Por ejemplo, el usuario “pedro” puede pedir un préstamo de 3 libros a la biblioteca “1” y otro préstamo de 5 libros a la biblioteca “2” al mismo tiempo (e inclusive puede tener 1 préstamo activo por cada biblioteca), pero no puede pedir otro préstamo a ninguna de las 2 anteriores hasta que devuelva los libros prestados. Por lo tanto, es importante que de cada préstamo se registre la fecha del mismo, y también la fecha de entrega (ésta se debe llenar sólo cuando el usuario efectivamente devuelve el libro)
En el caso de un usuario plus, si en el préstamo hay diferente tipo de libros será necesario registrarlo como 2 préstamos, y en ese caso tendrá 2 préstamos activos en una misma biblioteca.
Además, cuando un usuario plus solicita un préstamo hay que verificar que esté al día con el pago de su cuota de usuario plus.

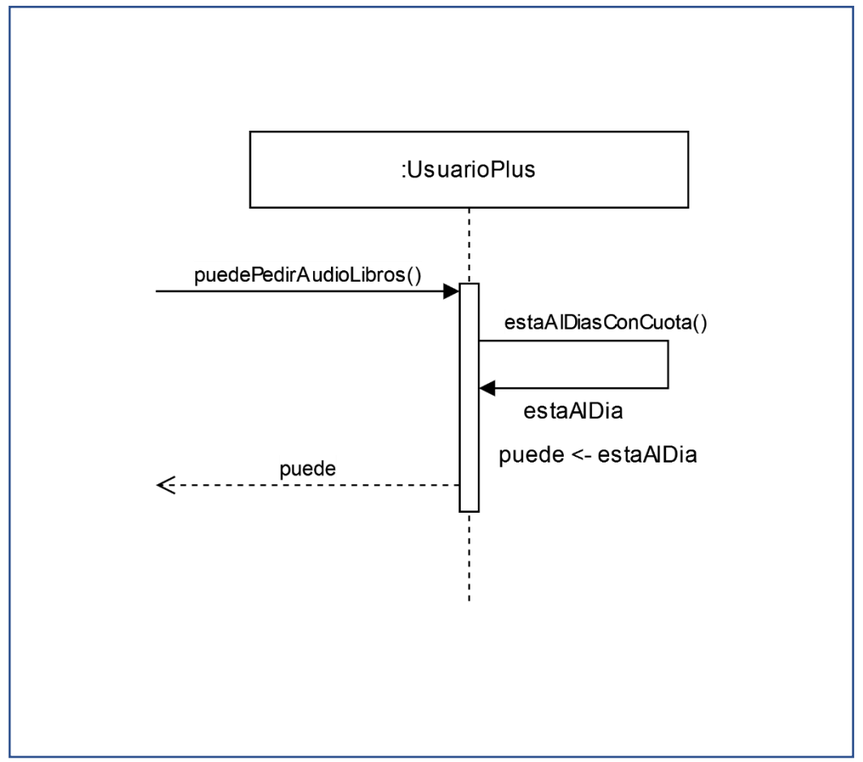


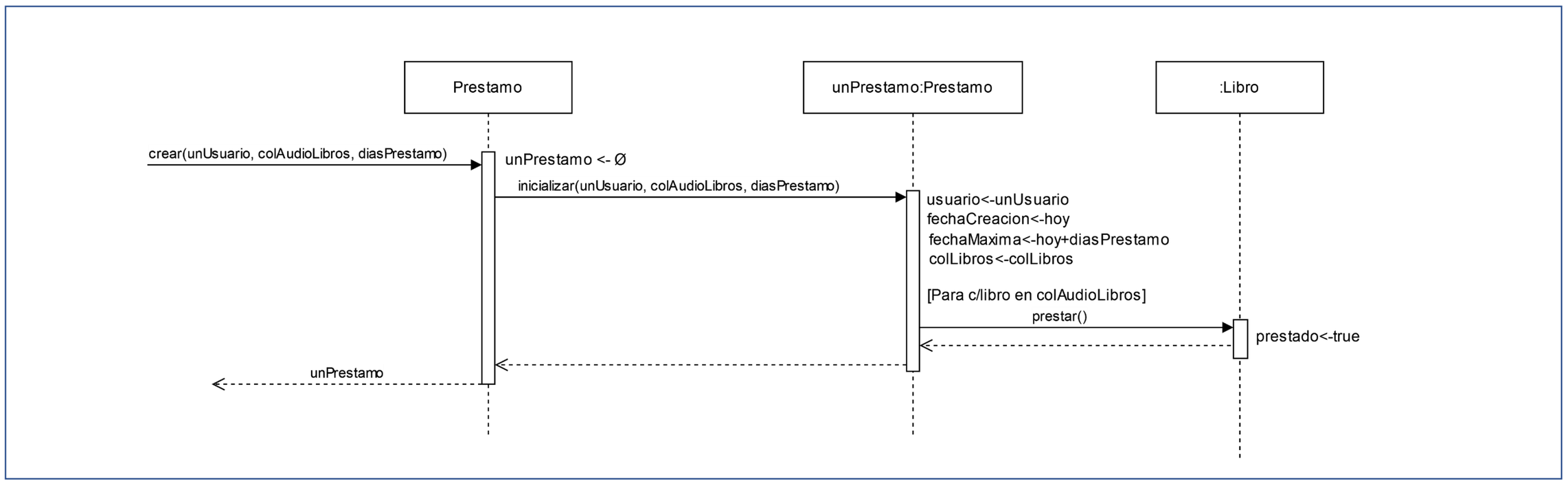
Diagramas de Secuencia

registrarPrestamo(unUsuario, unaColLibros)

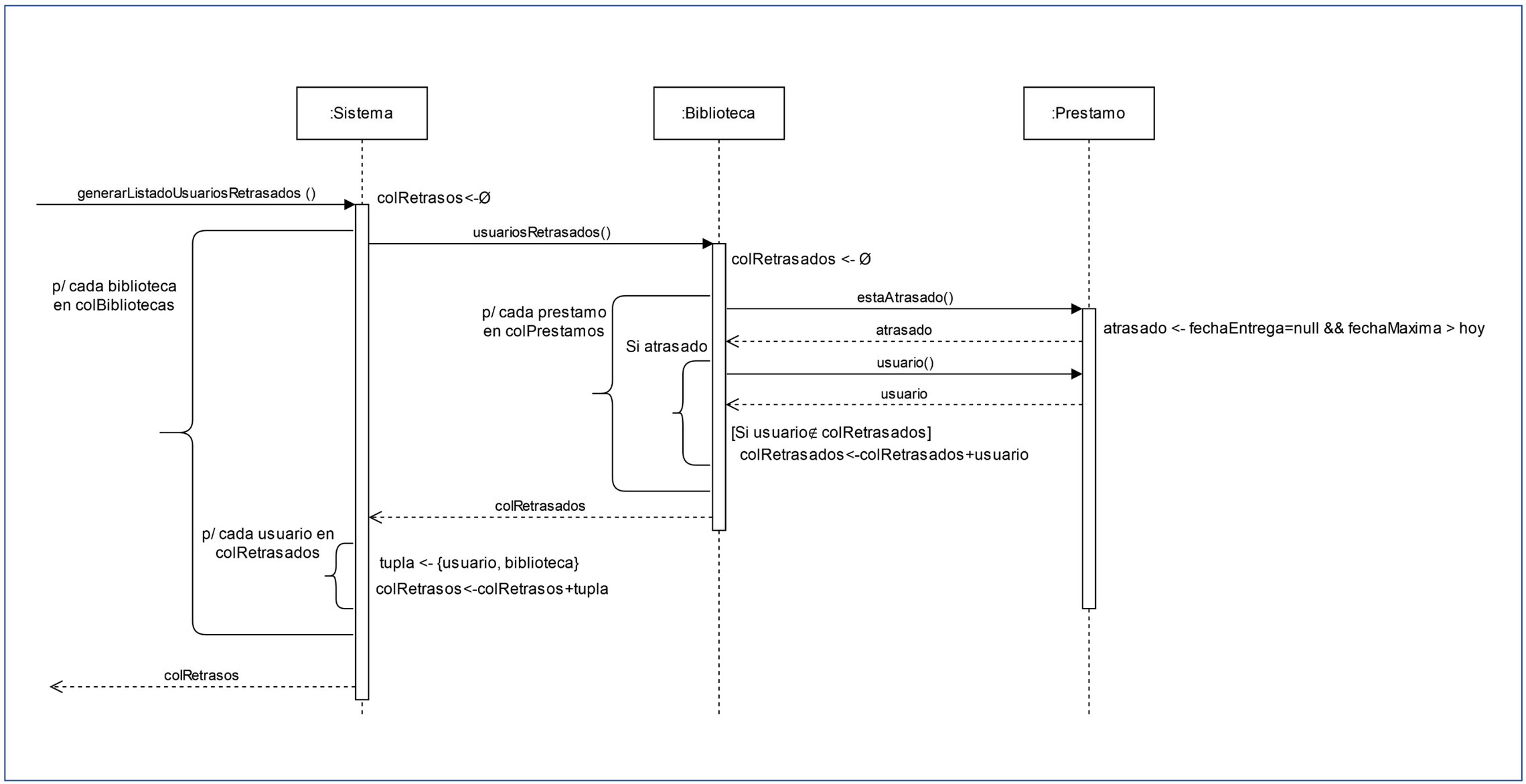



:Biblioteca
:Prestamo
tienePrestamoActivo(unUsuario)
tienePrestamos <- false
p/ cada prestamo
en colPrestamos
hasta
tienePrestmaos = true
estaActivo()
esActivo
esActivo<- fechaEntrage != nil
esDeUsuario(unUsuario)
esPropio
esPropio <-usuario = unUsuario
[Si esActivo && esPropio]
tienePrestamos <- true
tienePrestamos


:UsuarioPlus
puedePedirAudioLibros()
estaAlDiasConCuota()
estaAlDia
puede <- estaAlDia
puede


Prestamo
unPrestamo:Prestamo
:Libro
crear(unUsuario, colAudioLibros, diasPrestamo)
unPrestamo <- Ø
inicializar(unUsuario, colAudioLibros, diasPrestamo)
usuario<-unUsuario
fechaCreacion<-hoy
fechaMaxima<-hoy+diasPrestamo
colLibros<-colLibros
[Para c/libro en colAudioLibros]
prestar()
prestado<-true
unPrestamo

generarListadoUsuariosRetrasados()



devolverPrestamo(unUsuario)

El método solo devuelve prestamos de libros que no son audio libros. Si se crean prestamos de libros normales y audiolibros, con distinta fecha de vencimiento, es porque se da la opción de devolver primero uno y despúes otros.

Por lo tanto el metodo debe poder saber que devolver y para devolver audio libros se utiliza entonces devolverPrestamoAudioLibros(unUsuario).

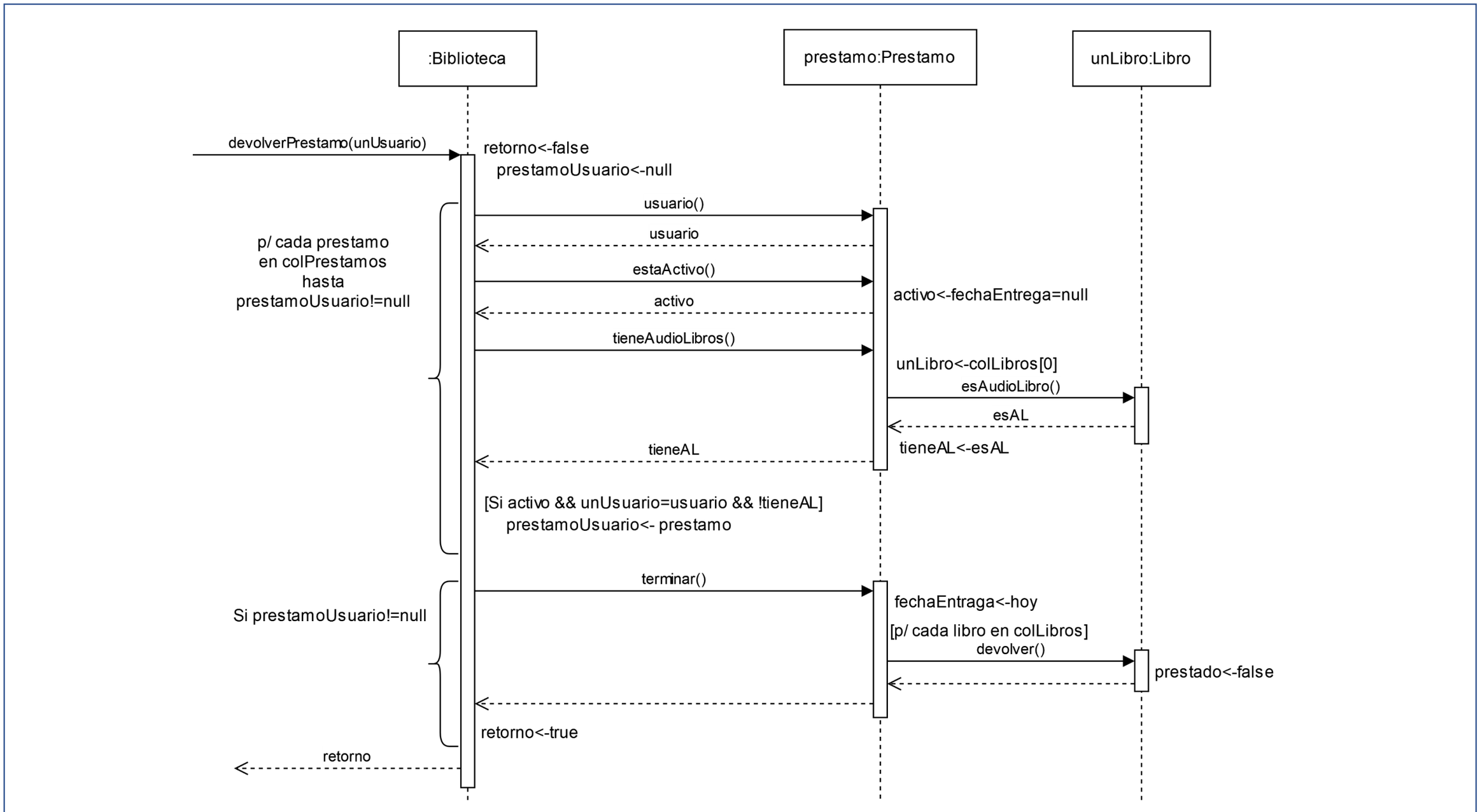
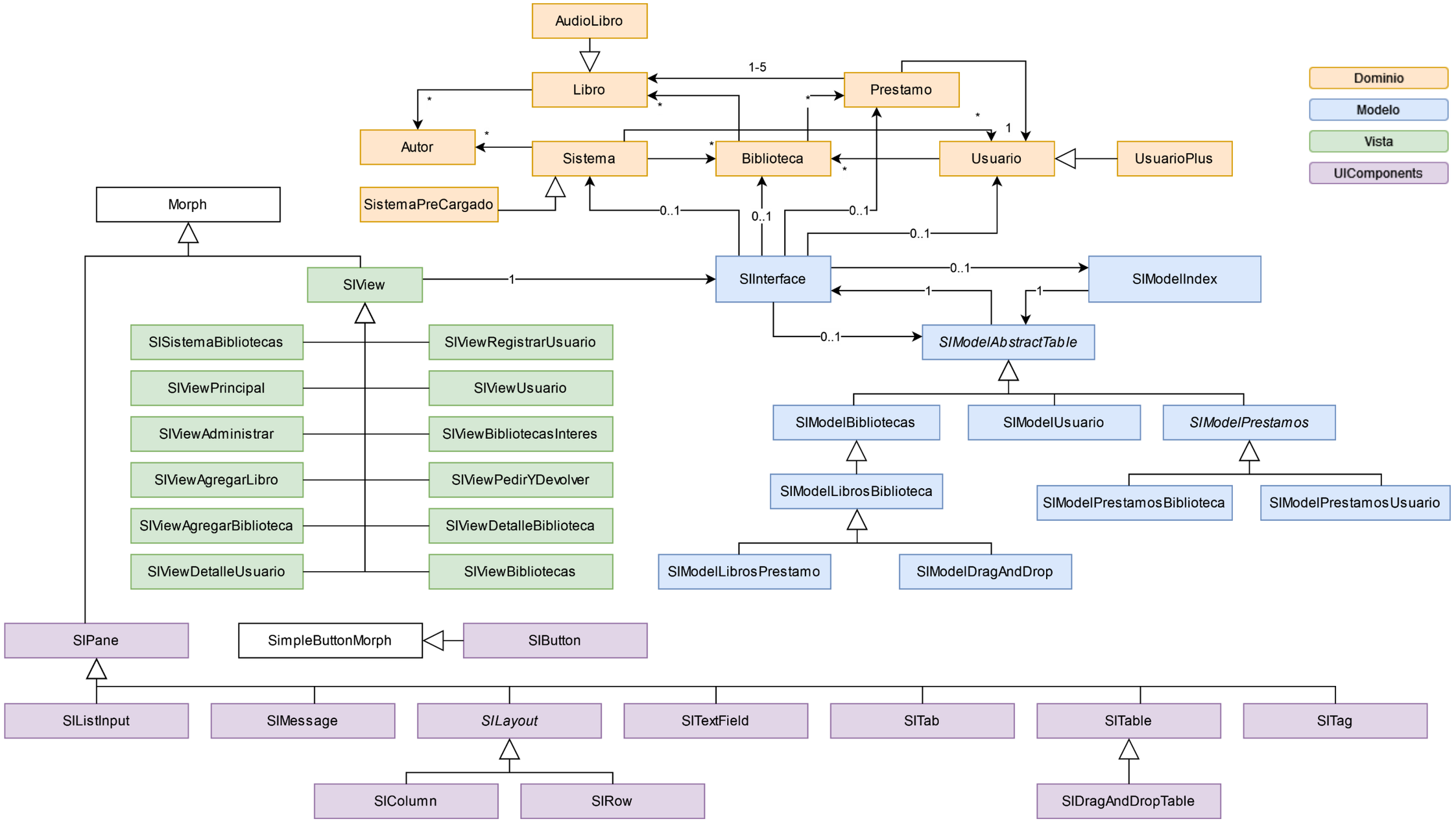


Diagram de clases Completo



Mensajes y Atributos de las clases

Clases del Dominio

Sistema

+agregarBiblioteca:

+agregarLibro:enLaBiblioteca:

+agregarUsuario:

+bibliotecas

+convertirANormal:

+convertirAPlus:

+existeDNIUsuario:

+existeIDBiblioteca:

+generarListadoUsuariosRegistrados

+loggin:password:

+obtenerUsuario:

+prestamosActivosDeUsuario:

+registrarAutoresDelLibro:

+usuarios

+initialize

Libro

-nombre

-editorial

-año

-prestado

+autores

+año

+devolver

+editorial

+esAudioLibro

+esPrestado

+prestar

+initializeNombre:editorial:año:autores:

Autor

-nombre

+nombre

+initializeNombre:

SistemaPreCargado

+cargarAutores:

+cargarLibros:

+initialize

Prestamo

-fechaCreacion

-fechaEntrega

-fechaMaxima

+cantidadLibros

+diasHastaVencimiento

+esDeUsuario:

+estaActivo

+estaRetrasado

+fecha

+fechaMaxima

+libros

+terminar

+tieneAudioLibros

+usuario

+initializeUsuario:libros:dias:

AudioLibro

+esAudioLibro

Biblioteca

-id

-diasPrestamo

-nombre

-representante

-telefono

+agregarLibro:

+devolverPrestamo:

+devolverPrestamoAudioLibro:

+direccion

+direccion:

+hasID:

+id

+libros

+librosDisponibles

+nombre

+nombre:

+prestamos

+prestamosActivos

+prestamosActivosDe:

+prestamosAtrasados

+registrarPrestamoUsuario:libros:

+removerLibro:

+representante

+representante:

+telefono

+telefono:

+tienePrestamoActivo:

+usuariosAtrasados

+initializeID:nombre:telefono:direccion:representante:

Usuario

-nombre

-apellido

-dni

-password

-diasSuspension

+apellido

+bibliotecasDeInteres

+bibliotecasDeInteres:

+dni

+esPlus

+estaSuspendido

+hasDNI:

+hasPassword:

+nombre

+password

+puedePedirAudioLibros

+tieneBibliotecasDeInteres

+initializeDNI:nombre:apellido:password:

UsuarioPlus

+esPlus

+estaAlDiaConCuota

+puedePedirAudioLibros

Clases de Modelo

SIInterface
+lib
+lib:
+loan
+loan:
+model
+model:
+modelIndex
+modelIndex:
+sys
+user
+user:
+initializeSistema:

SIModelAbstractTable
+cellCollumn:row:
+elementAt:
+iconFor:At:
+numberofRows
+refresh
+stringFor:at:
+initializeInterface:

SIModelUsuarios
+iconFor:at
+stringFor:at
+refresh

SIModelBibliotecas
+agregar
+refresh
+stringFor:at:

SIModelIndex
-index
+element
+elementChanged
+index
+model
+initializeModel:index:

SIModelLibrosBiblioteca
+agregar
+iconFor:at:
+refresh
+remove:
+removeAt:
+stringFor:at:

SIModelLibrosPrestamo
+refresh

SIModelPrestamosUsuario
+refresh

SIModelDragAndDrop
+addAll:
+array:
+dropElements:index:
+passengerAt:
+refresh
+removeAt:
+wantsDropElements:type:index:
+initializeArray:

SIModelPrestamosBiblioteca
+refresh

UIComponents

SIButton
-disabled
+disable
+enable
+handleEvent:
+initializeLabel:target:message:

SITag
+bad:
+clear
+good:
+initialize

SIPane
+initialize

SIDragAndDropTable
+addAll
+disable
+elements
+enable
+initializeModel:headers:

SITabs
-onTabChangedBlock
+addPage:label:
+onTabChanged:
+updatePageIndex:
+initialize

SILayout
+initialize
+initializeOrientation

SIListInput
-list
-textField
-model
+agregar
+elements
+elements:
+eliminar
+initialize

SITable
-table
-doubleClickBlock
-clickBlock
+eliminar
+model
+model:
+onClick:
+onDoubleClick:
+selected
+selectedIndex
+initializeModel:headers:

SIColumn
+initializeOrientation

SITextField
-border
-textField
+accepted
+rejected
+text
+text:
+initialize
+initializeTag:

SIMessage
+initializeContents:

SIRow
+initializeOrientation

Clases Vista

SIView

+initializeInterface:

SIViewAgregarBiblioteca

-tfNombre
-tfDireccion
-tfTelefono
-tfRepresentante
-tfID
-tag

+actionAgregar
+initializeInterface:

SIViewAgregarLibro

-tfNombre
-tfEditorial
-tfAño
-lfAutores

+actionAgregar
+initializeInterface:

SISistemaBibliotecas

+initializeInterface:

SIViewUsuario

-tag

+actionCambiarBibliotecasInteres
+actionPedirYDevolver
+actionSolicitarNormal
+actionSolicitarPlus
+initializeInterface:

SIViewUsuarioBibliotecasDeInteres

-outTable
-tag

+actionAceptar
+actionCerrarSesion
+initializeInterface:

SIViewAdministrar

-tablaBiblioteca
-tablaUsuario

+actionAgregarBiblioteca
+initializeInterface:

SIViewPedirYDevolver

-outTable
-inTable
-tag
-btnPedir
-btnDevolverLibros
-btnDevolverAudioLibros
-prestamoLibro
-prestaniAudioLibro
-labelCanasta
-lbLeftDays
-lbLeftDaysAL

+actionDevolverAudioLibros
+actionDevolverLibros
+actionPedirPrestamo
+prepararDevolverAudioLibros
+prepararDevolverLibros
+prepararPedirPrestamo
+refresh
+initializeInterface:

SIViewDetalleBiblioteca

-ftRrepresentante
-ftDireccion
-ftTelefono
-ftNombre
-tablaLibros

+actionAgregarLibro
+actionConfirmarDatos
+actionDarDeBaja
+initializeInterface:

SIViewPrincipal

-tfUsuario
-tfContraseña
-tag

+actionAdministrar
+actionIniciarSesion
+actionRegistrarse
+esContraseñaVacio
+esDNIInvalido
+esUsuarioVacio
+initializeInterface:

SIViewRegistrarUsuario

-tfRepetirPass
-tfPass
-tfDNI
-tfApellido
-tfNombre
-tag

+actionRegistrar
+esApellidoVacio
+esDNIInvalido
+esNombreVacio
+esPassVacio
+passEquals
+initializeInterface:

SIViewUsuarioBibliotecas

+initializeInterface:

SIViewDetalleUsuario

+initializeInterface:

Implementación

https://github.com/sebaiova/POO-TPFinal

Instalación

En el playground de Pharo ejectuar

```
Metacello new
        githubUser: 'sebaiova' project: 'POO-TPFinal' commitish: 'master' path: 'src';
        baseline: 'SI';
        load.
```

Ejecución

Con sistema vacío

```
SISistemaBibliotecas run
```

Con datos precargados

```
SISistemaBibliotecas runPreloaded
```

Usuario Test

Usuario: 39062823

Password: 1234

Tiene un préstamo vencido.

Todas las bibliotecas tienen el mismo set de libros (libros distintos, copias).