

## Disponibilidad

Para poder mejorar la disponibilidad del Sistema de mensajería, se implementaron las siguientes tácticas:

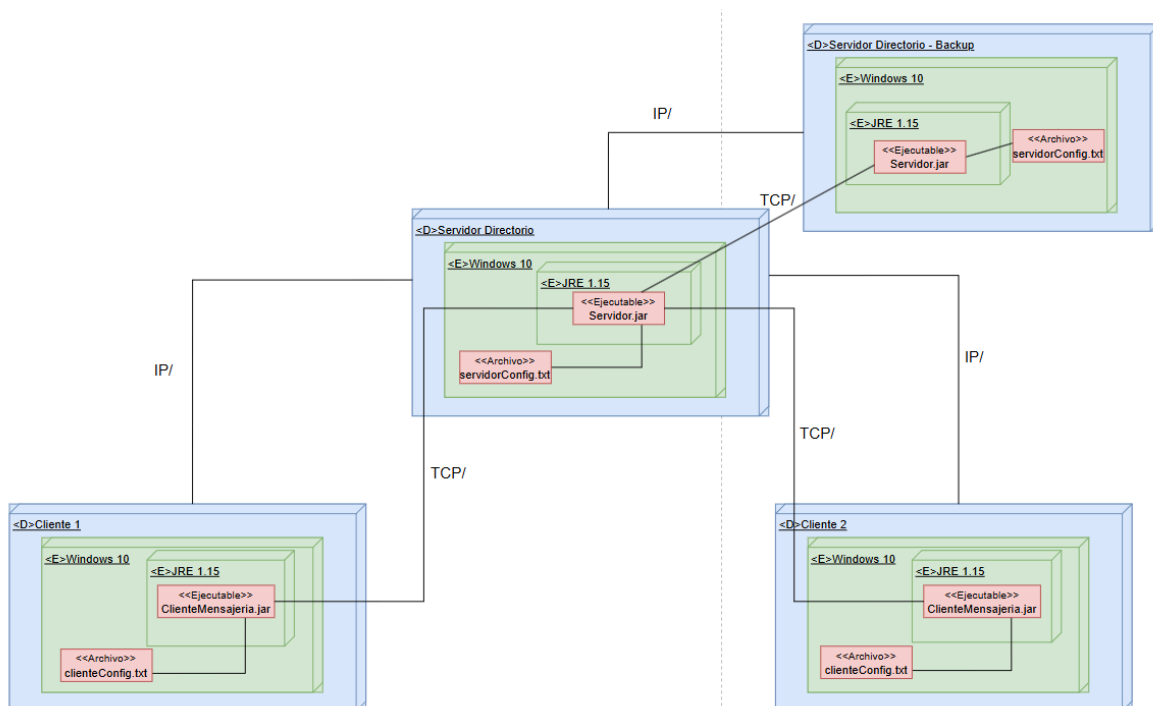
### Detección de fallas:

- Heartbeat: Para detectar una eventual falla o caída del servidor primario, este envía un pulso periódico al servidor secundario. Si este último no recibe un pulso en más de 5 segundos, asume que su contraparte falló.

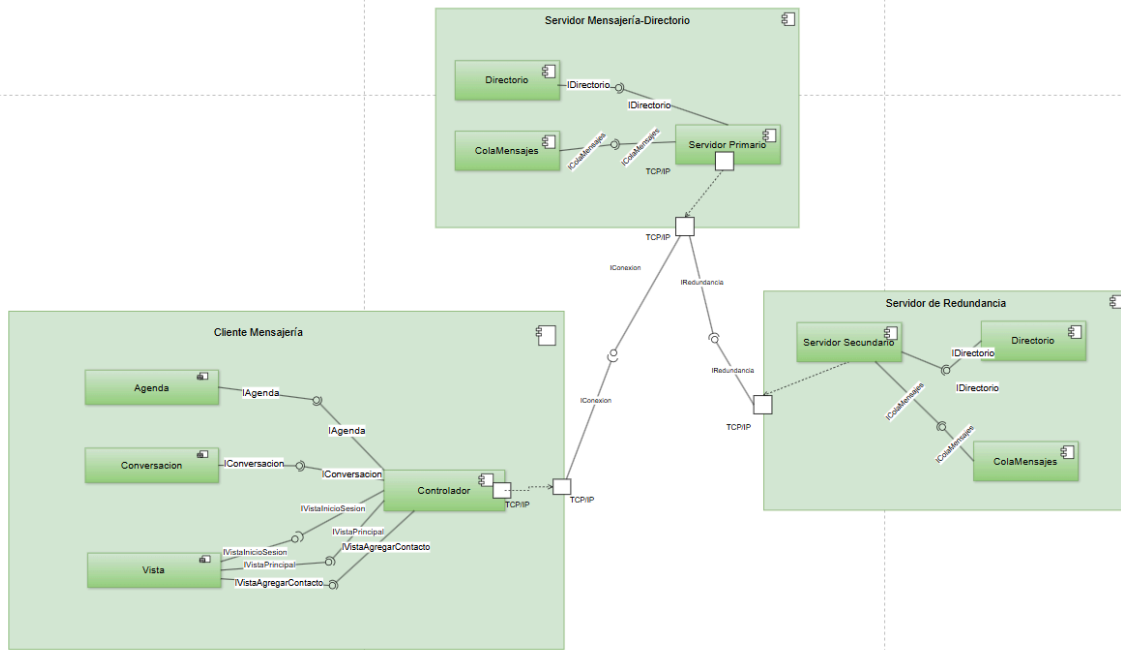
### Recuperación de fallas:

- Redundancia Pasiva: Se agrega al servidor la capacidad de conectarse con una segunda instancia que funciona como backup. El mismo servidor cuenta con la lógica para detectar si se debe comportar como instancia primaria o de respaldo. Esto es debido a que en la redundancia pasiva solo un servidor recibe los mensajes del resto de los componentes. El servidor secundario mantiene continuamente su estado actualizado con respecto al principal.
- Reintento: Al fallar cualquier conexión, envío o recepción de mensajes con el servidor, los sistemas cliente realizan un reintento con el mismo servidor antes de probar hasta dos veces con el de respaldo (si es que lo hubiera).
- Resincronización de estado: Si por algún motivo el servidor principal falla, se puede volver a iniciar. En dicho caso, él mismo se encargará de recuperar su estado anterior aprovechando la otra instancia implementada en la redundancia pasiva.

## Diagrama de Despliegue

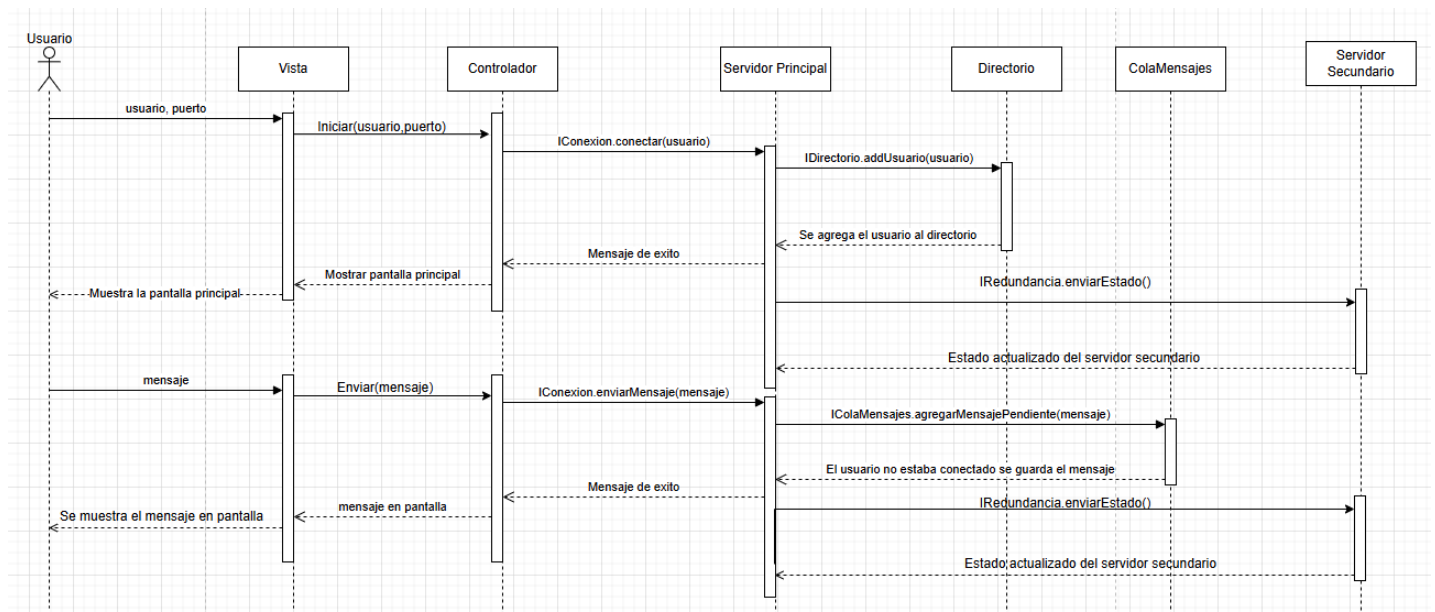


## Diagrama de componentes

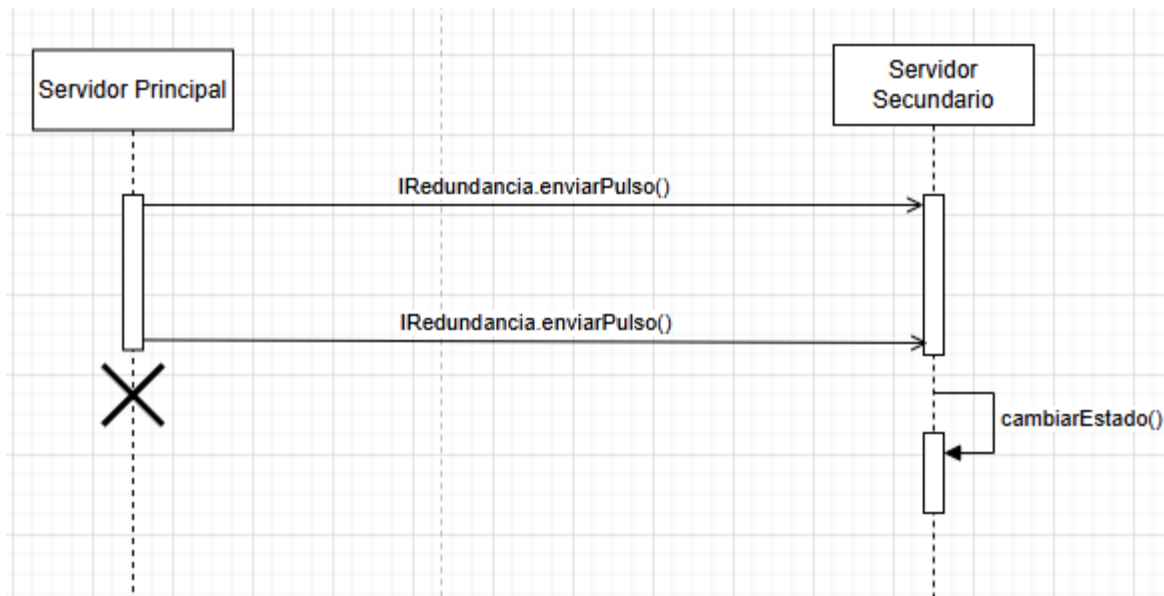


## Diagrama de secuencia

### Redundancia pasiva

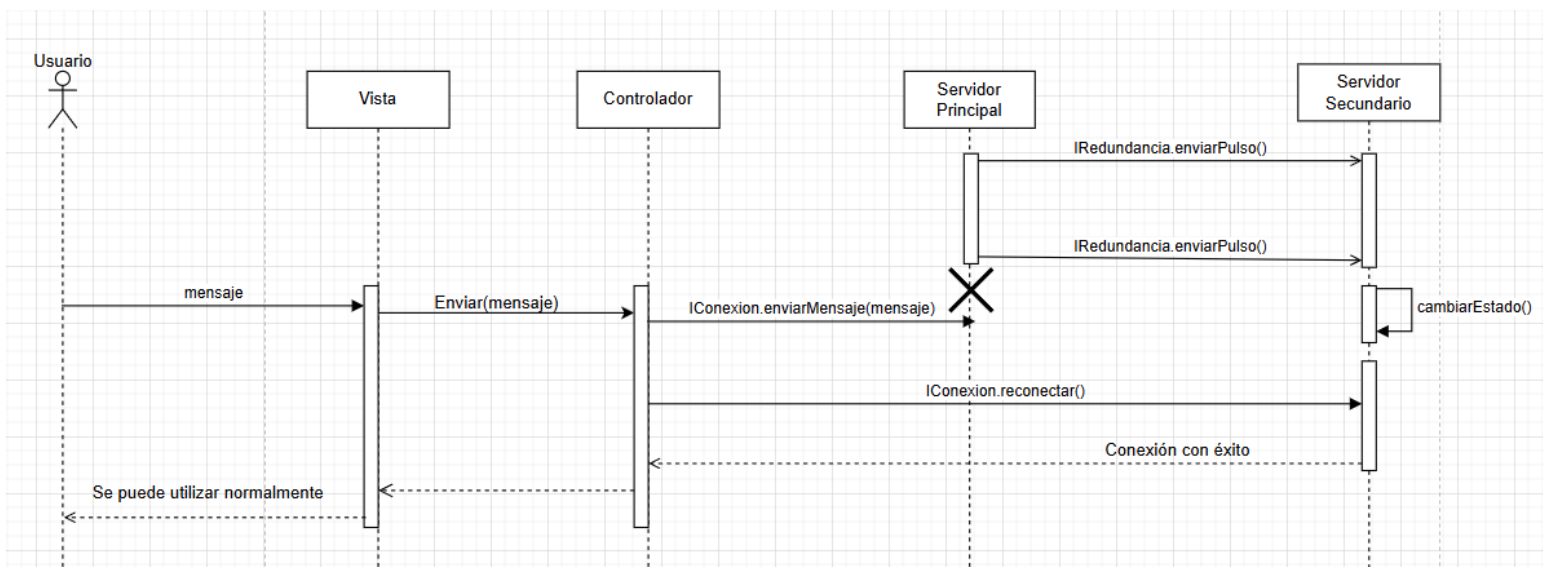


## Resincronización de estado y Heartbeat



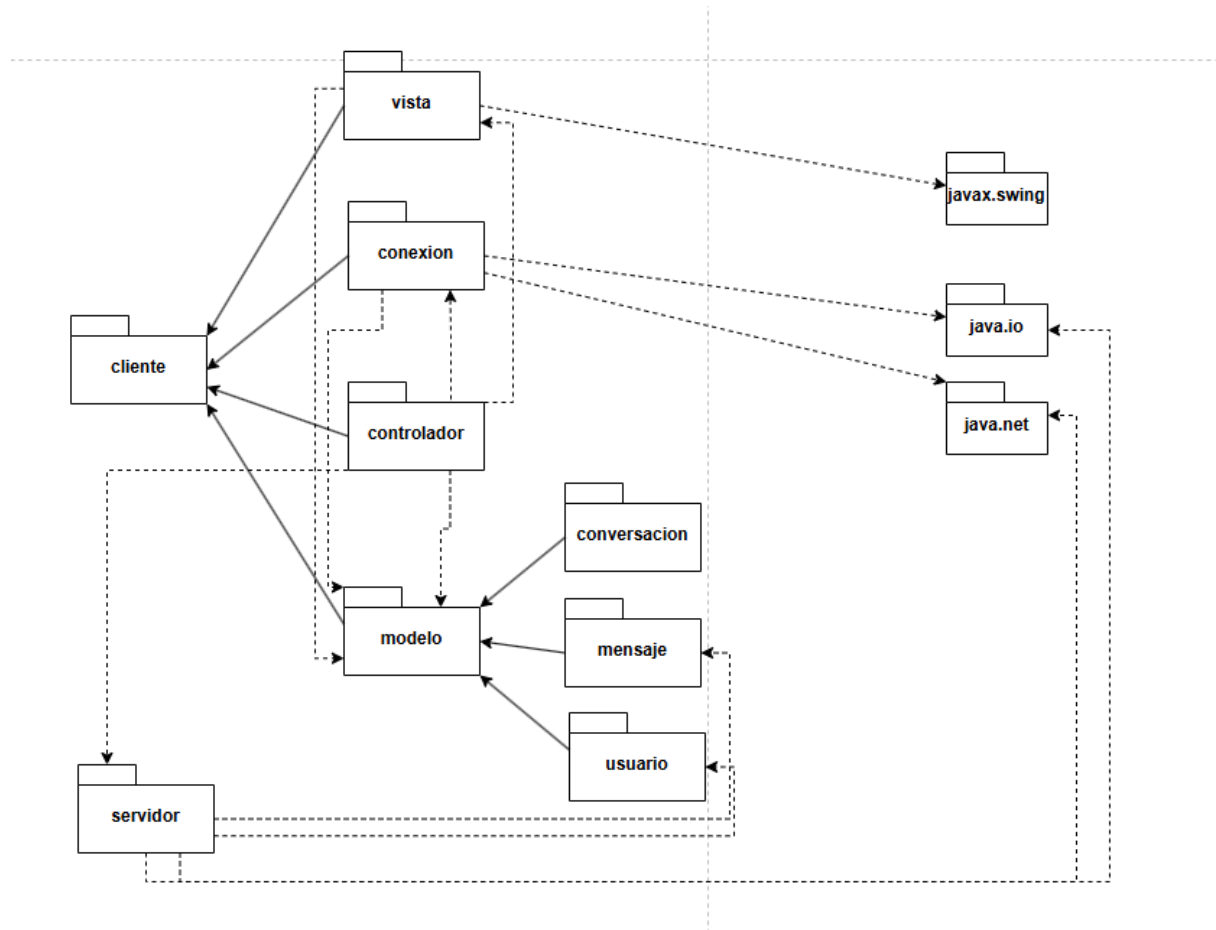
## Reintento

*El usuario ya estaba conectado*

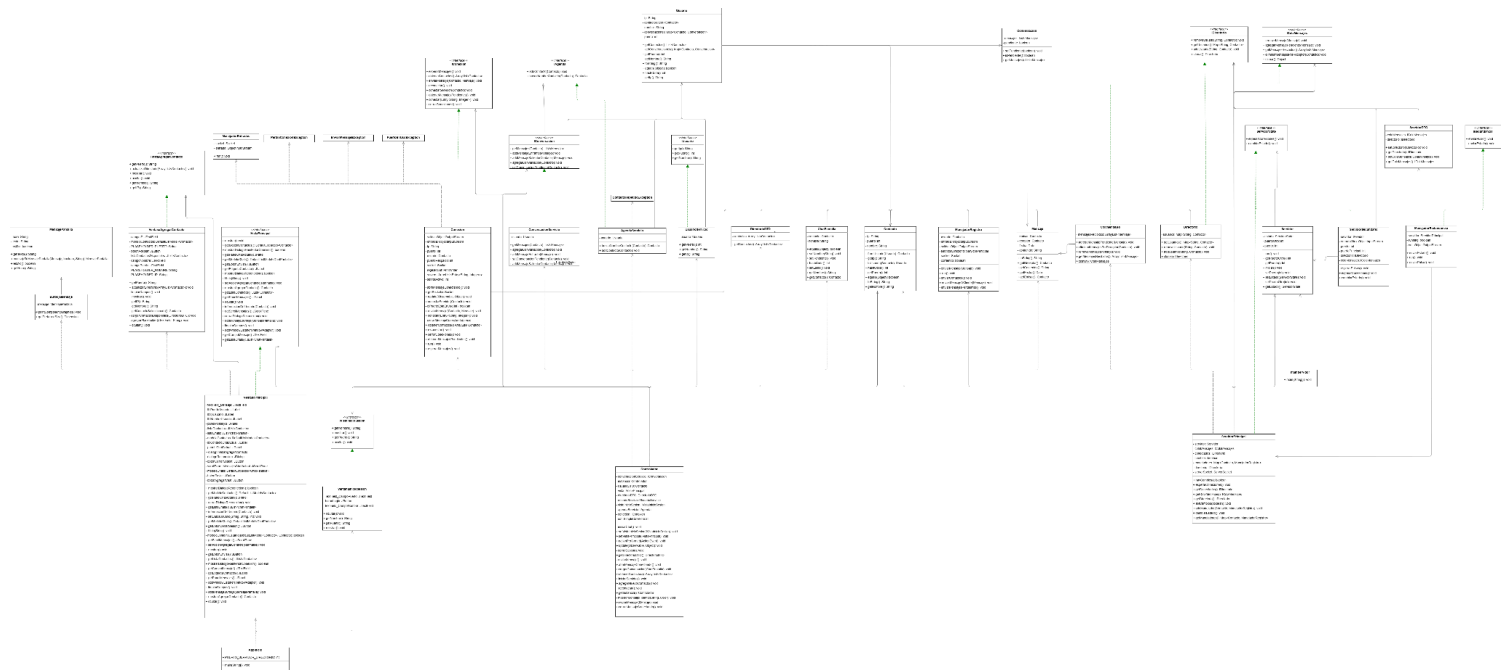


## Diagrama de paquetes

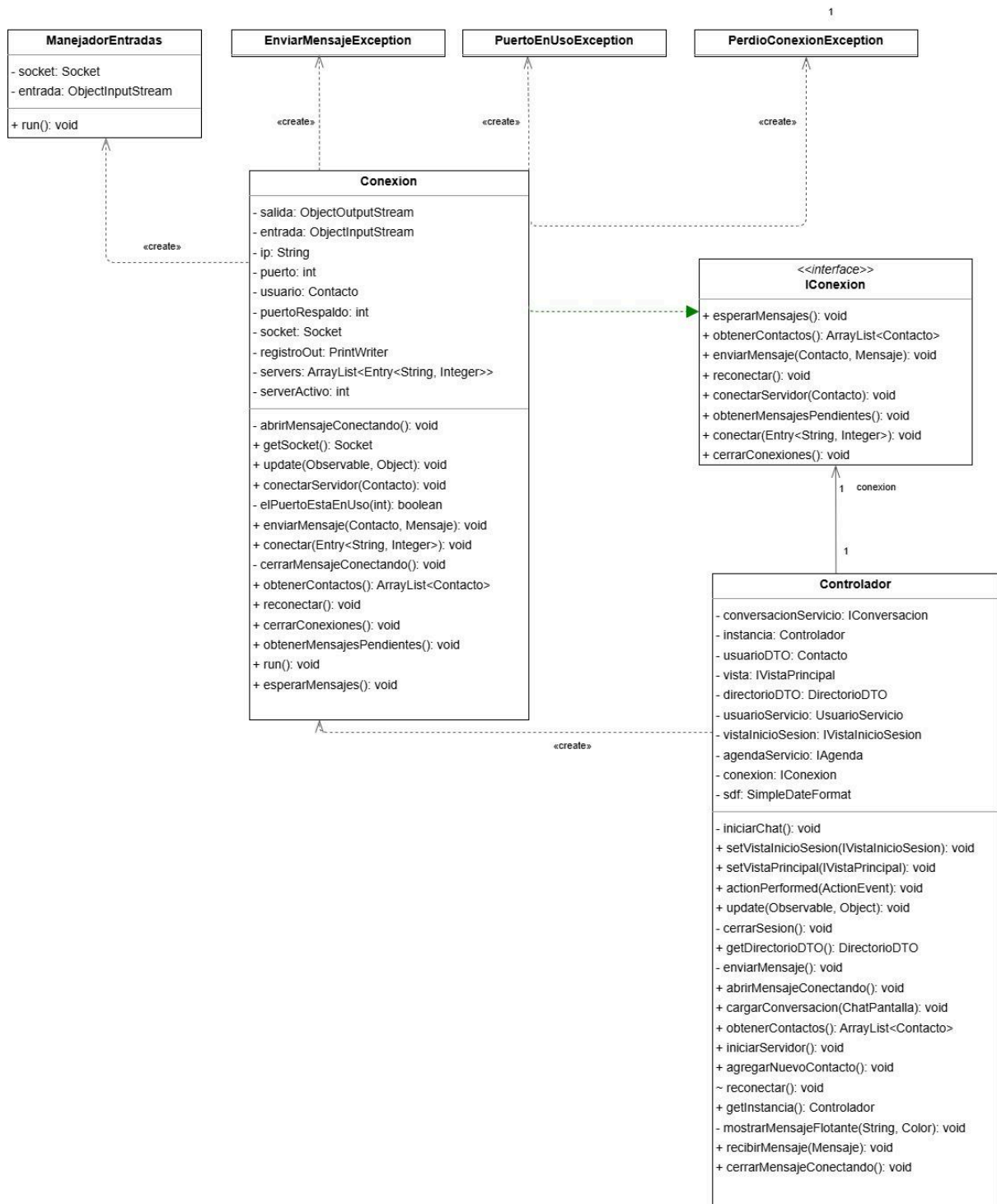
*No sufre cambios*



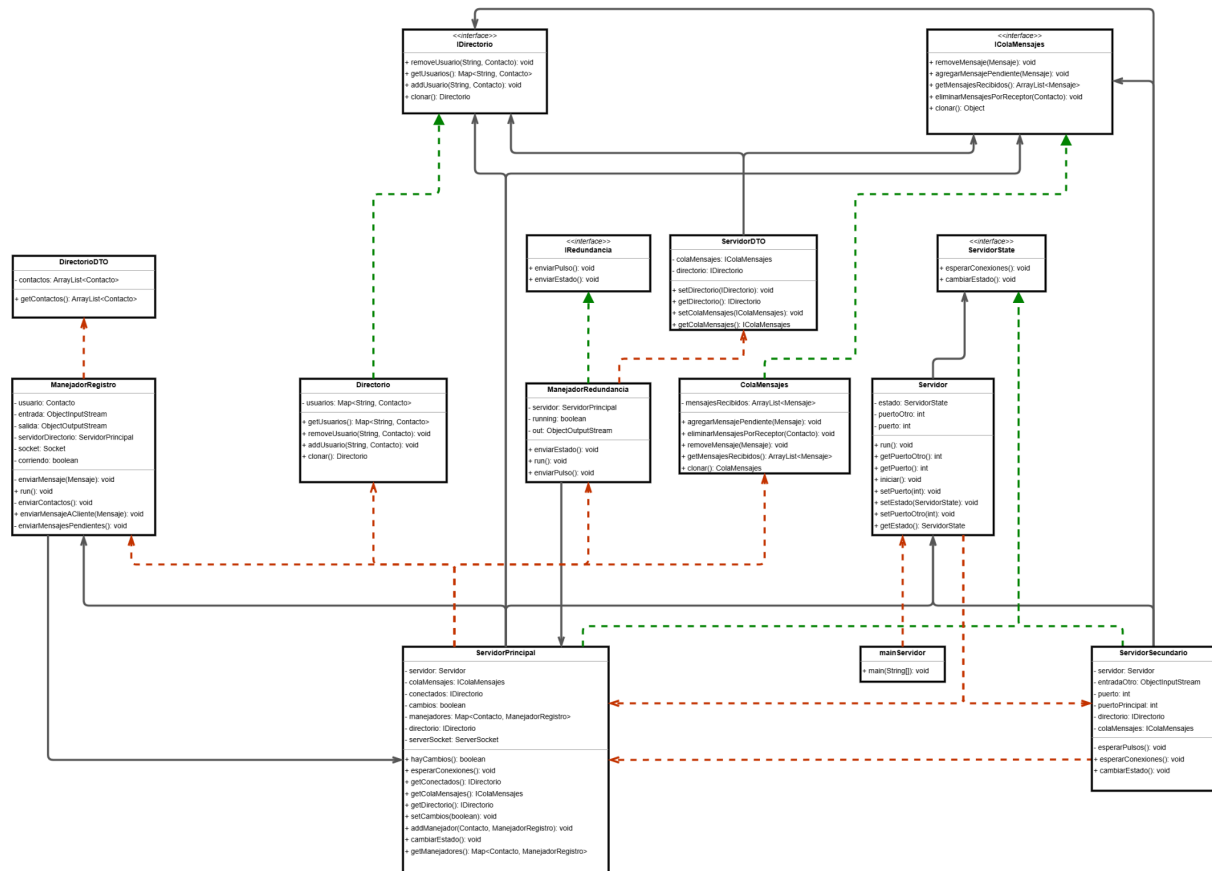
**Diagrama de clases:** [drawio](#)



## Clases que se vieron modificadas



## Clases Servidor: [drawio](#)



## Interfaces de Usuario

Configurar Usuario

Nombre de Usuario

Puerto

Aceptar

App de Mensajería - Usuario: usuario [Ip: 127.0.0.1] Puerto: 8081

Datos del Usuario:  
Nombre: usuario  
IP: 127.0.0.1  
Puerto: 8081

Contactos

Nuevo Contacto Iniciar Chat

Chats

Cerrar Sesión

Bienvenido

Nombre: usuario

IP: 127.0.0.1

Puerto: 8081

## Contactos

--

## Nuevo Contacto

[Iniciar Chat](#)

## Chats

--

Cerrar Sesión



# Bienvenido

App de Mensajería - Usuario: usuario [ip: 127.0.0.1] Puerto: 8081

-- □ X

Datos del Usuario:

Nombre: usuario

IP: 127.0.0.1

Puerto: 8081

Contactos

Nuevo Contacto

Iniciar Chat

Chats

Cerrar Sesión

Nuevo contacto

X

Agregar manualmente:

Nombre:

Usuario123

IP

192.168.0.1

Puerto

8008

Agregar Contacto

Contactos Disponibles:

Seleccionar Contacto

Nombre: usuario

IP: 127.0.0.1

Puerto: 8081

## Contactos

**Nuevo Contacto**

[Iniciar Chat](#)

## Chats

--

Cerrar Sesión

Nuevo contacto		X
<b>Agregar manualmente:</b>		<b>Contactos Disponibles:</b>
Nombre: <input type="text" value="Usuario123"/>		
IP <input type="text" value="192.168.0.1"/>		
Puerto <input type="text" value="8008"/>		
<input type="button" value="Agregar Contacto"/>		
		<input type="button" value="Seleccionar Contacto"/>

### Nuevo contacto

Agregar manualmente:

Nombre:

Usuario123

IP

192.168.0.1

Puerto

8008

[Agregar Contacto](#)

**Contactos Disponibles:**

### Seleccionar Contacto



