

Los Atuendos

## ACTIVIDAD 4

Patrones de diseño de software

Los Atuendos

Profundización

Cristian Andres Pulido Cristancho

Ucompensar

Ingeniería de software

Bogotá, Colombia

2025-2

## **Patrones de creación y patrones estructurales.**

### **INDICE**

Patrones de creación y patrones estructurales.....	2
Patrones de diseño identificados.....	3
Justificación general.....	3
Patrones de secuencia .....	5
Patrón Factory Method Creación de una Prenda .....	5
Patrón Singleton: Acceso a la Única Instancia del Negocio .....	5
Patrón Composite: Cálculo del Valor de Alquiler de un Traje Compuesto .....	6
Patrón Facade: Devolución de Prendas y Envío a Lavandería .....	7

### Patrones de diseño identificados

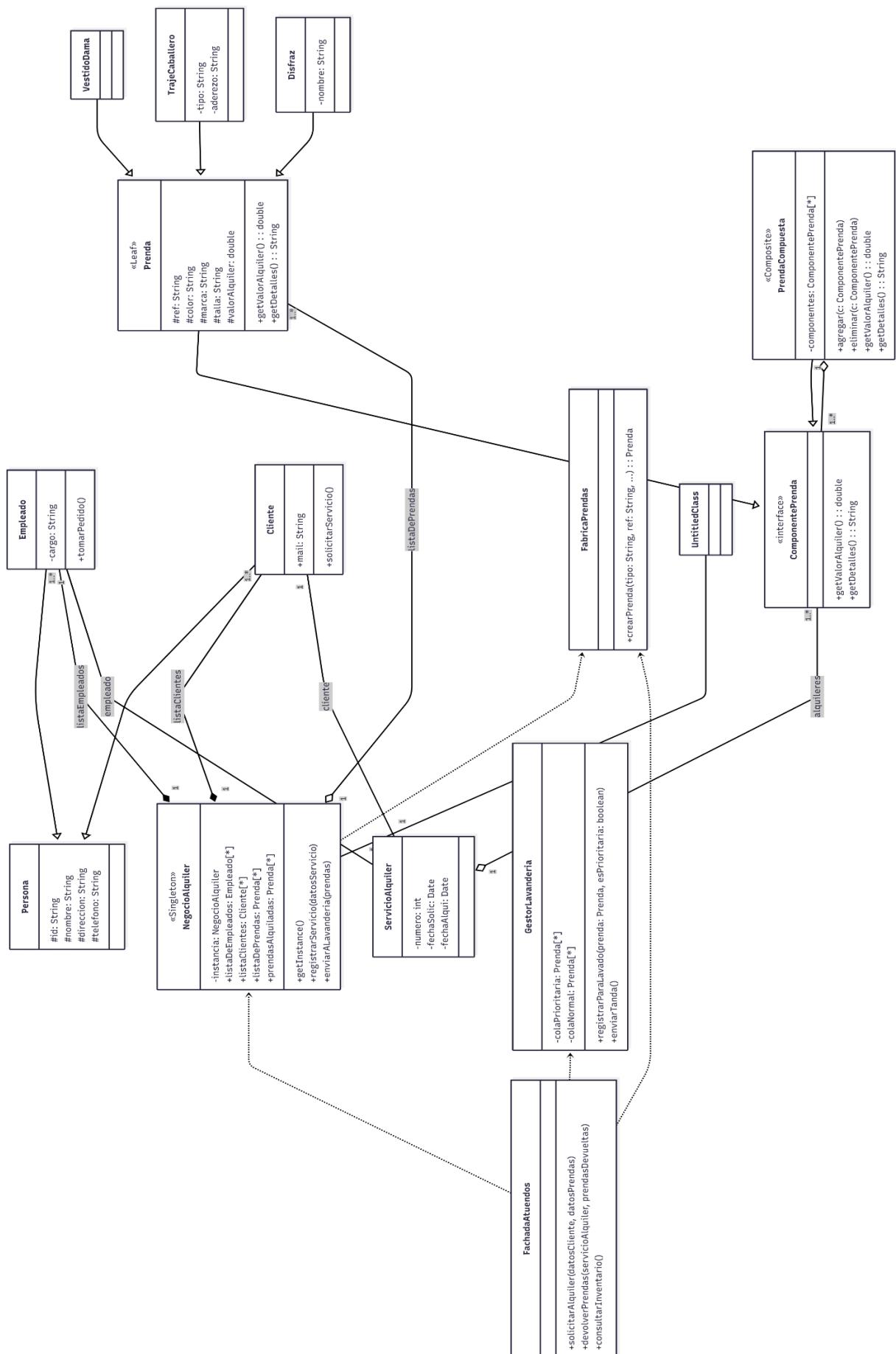
Patrón a Utilizar	Tipo (Creación o Estructural)	¿Por qué contribuye a mejorar el diseño?
<b>Factory Method</b>	Creación	Permite crear objetos de tipo Prenda (VestidoDama, TrajeCaballero, Disfraz) sin acoplar la lógica del negocio a las clases concretas. Facilita la extensión si en el futuro se agregan nuevos tipos de prendas sin modificar el código existente.
<b>Singleton</b>	Creación	Garantiza que existe una única instancia de NegocioAlquiler que gestione los registros de prendas, clientes y empleados. Evita inconsistencias de datos y facilita el control centralizado de la información.
<b>Composite</b>	Estructural	Permite tratar de forma uniforme las prendas que están compuestas por varias piezas (como los vestidos de dama con varias partes). Mejora la flexibilidad del sistema al manejar colecciones jerárquicas de objetos.
<b>Facade</b>	Estructural	Proporciona una interfaz unificada para acceder a las operaciones del sistema (registro, consultas, envío a lavandería). Reduce la complejidad del sistema para el usuario y mejora la mantenibilidad.

### Justificación general

El sistema “Los Atuendos” está compuesto por diversas entidades interrelacionadas, tales como *Clientes, Empleados, Prendas y Servicios de Alquiler*. Dada esta complejidad, la aplicación de patrones de diseño creacionales resulta esencial para separar adecuadamente la lógica de instanciación de la lógica de negocio, facilitando así la mantenibilidad y escalabilidad del sistema.

Por otro lado, la incorporación de patrones de diseño estructurales contribuye a lograr una arquitectura más modular y flexible, permitiendo incorporar nuevas funcionalidades o clases sin afectar las ya existentes. Esta característica es especialmente relevante en sistemas de gestión de alquileres, donde la evolución continua de los procesos y requerimientos exige un diseño adaptable y sostenible en el tiempo.

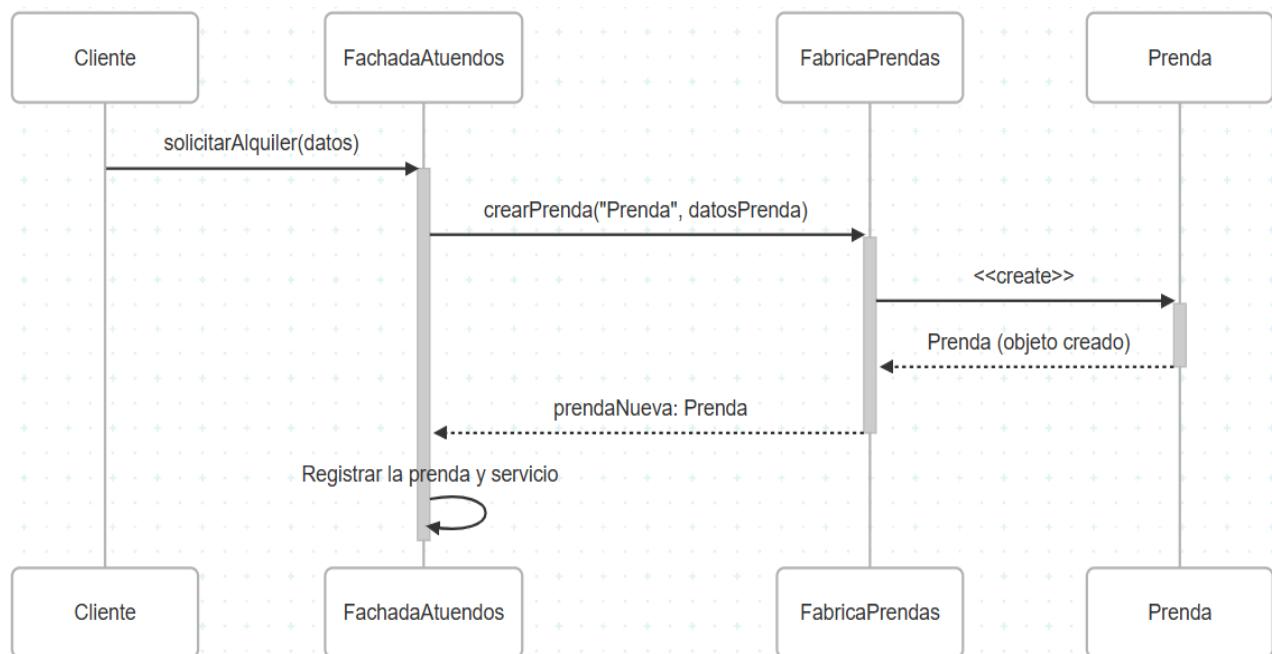
## Los Atuendos



### Patrones de secuencia

#### Patrón Factory Method: Creación de una Prenda

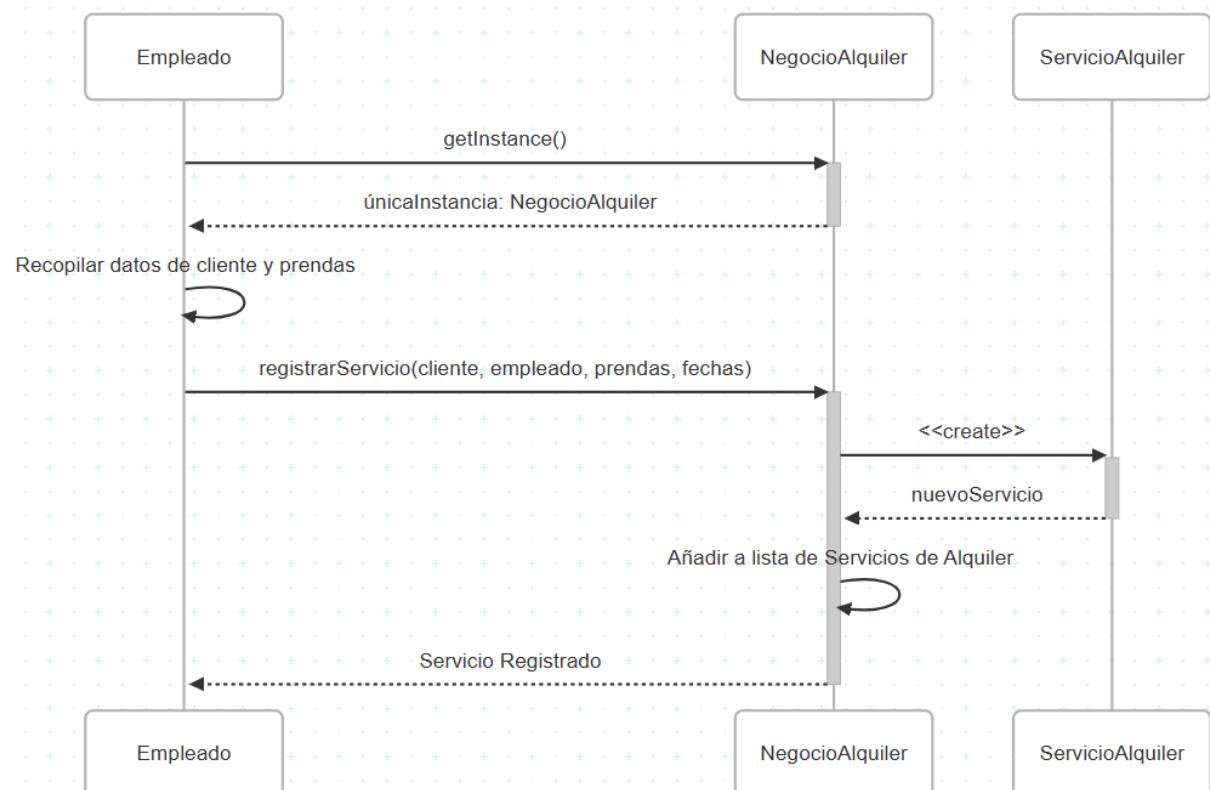
El cliente solicita un nuevo tipo de prenda. La **FabricaPrendas** se encarga de instanciar la clase concreta (VestidoDama) sin que la lógica del negocio (NegocioAlquiler o FachadaAtuendos) necesite conocer los detalles de esa clase específica.



#### Patrón Singleton: Acceso a la Única Instancia del Negocio

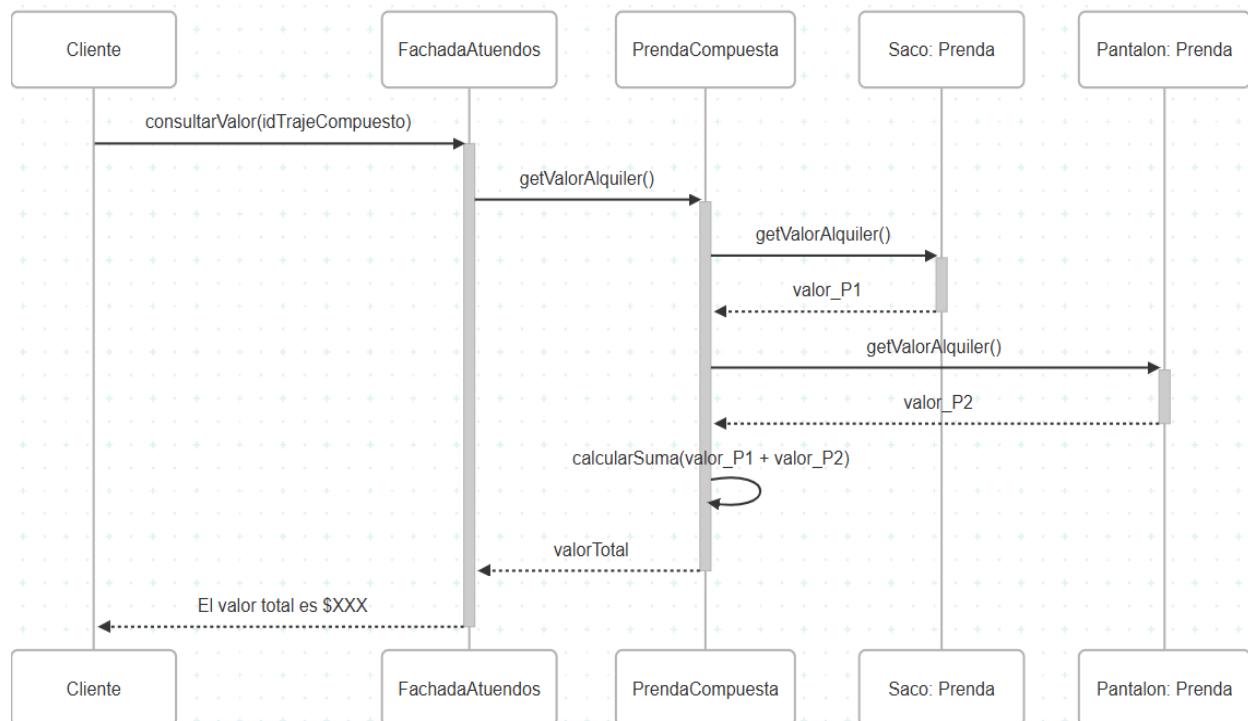
Este es para cuando se requiere acceso a la gestión centralizada (por ejemplo, para registrar un servicio de alquiler), este asegura que siempre se trabaje con la única instancia de **NegocioAlquiler**.

## Los Atuendos



### Patrón Composite: Cálculo del Valor de Alquiler de un Traje Compuesto

Un traje de caballero puede ser una “PrendaCompuesta” de saco, pantalón y aderezo, donde cada parte es un objeto Prenda individual. El cliente puede consultar el valor total del alquiler de la composición, y el *Composite* (PrendaCompuesta) delega la suma del valor a sus *prendas*.



## Los Atuendos

### Patrón Facade: Devolución de Prendas y Envío a Lavandería

El cliente interactúa con la **FachadaAtuendos** para simplificar la secuencia de devolución, que involucra actualizar el servicio, registrar la prenda para lavado, y enviarla al **GestorLavanderia**.

