

# ANEXO III – ESPECIFICACIÓN DE DISEÑO



VNiVERSiDAD  
D SALAMANCA

Trabajo de Fin de Grado  
Grado en Ingeniería Informática

Septiembre 2023



# ÍNDICE

1.	Introducción.....	1
2.	Ámbito de la aplicación.....	3
3.	Diseño.....	5
3.1.	Tecnologías elegidas .....	5
3.2.	Subsistemas de diseño .....	5
3.3.	Diagrama de clases de diseño .....	6
3.4.	Casos de uso de diseño .....	6
3.5.	Diseño de la base de datos .....	11
3.6.	Modelo de despliegue .....	12
4.	Implementación .....	13
4.1.	Subsistemas de implementación con componentes .....	13
4.2.	Clases incluidas en cada componente .....	14



## LISTA DE ILUSTRACIONES

Ilustración 1 - Subsistemas de diseño .....	6
Ilustración 2 - Diagrama de clases de diseño .....	6
Ilustración 3 - Diagrama de estados de CU - 001 .....	7
Ilustración 4 - Diagrama de secuencia de diseño de CU - 001 .....	7
Ilustración 5 - Diagrama de estados de CU - 002 .....	7
Ilustración 6 - Diagrama de secuencia de diseño de CU - 002 .....	8
Ilustración 7 - Diagrama de estados de CU - 003 .....	8
Ilustración 8 - Diagrama de secuencia de diseño de CU - 003 .....	8
Ilustración 9 - Diagrama de estados de CU - 004 .....	9
Ilustración 10 - Diagrama de secuencia de diseño de CU - 004 .....	9
Ilustración 11 - Diagrama de estados de CU - 005 .....	9
Ilustración 12 - Diagrama de secuencia de diseño de CU – 005.....	10
Ilustración 13 - Diagrama de estados de CU - 006 .....	10
Ilustración 14 - Diagrama de secuencia de diseño de CU - 006 .....	10
Ilustración 15 - Diseño de datos de la tabla item .....	11
Ilustración 16 - Diseño de datos de la tabla usuario .....	11
Ilustración 17 - Diagrama de despliegue.....	12
Ilustración 18 - Diagrama de subsistemas con componentes de implementación .....	13
Ilustración 19 - Clases por cada componente.....	14



# 1. INTRODUCCIÓN

En este anexo se tratará la segunda parte del proceso de Ingeniería del Software para el desarrollo del proyecto. Las fases a tratar serán las de Diseño e Implementación, que se enmarcan ya en la parte del dominio de la solución, es decir, se trata el problema a un nivel más bajo, más alejado del planteamiento abstracto inicial y más cercano a cómo va a ser realmente implementado el sistema.

Se tomará como punto de entrada para hacer toda la documentación de diseño e implementación todo lo obtenido en la obtención de requisitos y análisis en “Anexo I – Especificación de Requisitos del Software”.





## 2. ÁMBITO DE LA APLICACIÓN

En este apartado se resumirán brevemente los aspectos más relevantes obtenidos en el “Anexo I – Especificación de Requisitos del Software”.

Los objetivos principales del sistema son la obtención de catálogos de ropa de hombre y mujer, la búsqueda y ordenación de los resultados mostrados por la aplicación según distintos criterios y la gestión de usuarios. Son muy relevantes pues sustentan los tres pilares que debe tener la aplicación, que son la obtención de los productos, una forma de visualizarlos y la posibilidad de tener cuentas de usuario y gestionarlas.

Los requisitos funcionales o casos de uso obtenidos son funcionalidades que tiene la aplicación y hay distintos actores que interaccionan con ella. Los actores son, Usuario Registrado, que hereda de Usuario Anónimo. Estos actores son los usuarios más comunes, siendo usuario anónimo un tipo de usuario que navega por la página web pero no está registrado. Por último, también está el actor Sistema, que se encarga de realizar tareas automáticas. Los casos de uso principales están agrupados en dos paquetes, Gestión de Usuarios y Gestión de Prendas. El paquete de Gestión de Usuarios contiene Iniciar Sesión y Crear Cuenta de Usuario como casos de uso y el paquete Gestión de Prendas tiene Obtener Catálogo, Ver todas las prendas, Filtrar prendas y Ver Prenda como casos de uso.



## 3. DISEÑO

La fase de diseño es la tercera etapa del desarrollo de un sistema en la Ingeniería del Software. Se obtiene como punto de entrada los diagramas resultantes de la fase de análisis, sobre los que se sigue refinando y trabajando sobre ellos. Esta etapa da comienzo al dominio de la solución, que se encuentra más cerca de la implementación real del sistema que de la idea abstracta planteada en la etapa de obtención de requisitos.

Lo que se tratará en esta etapa son el establecimiento de la tecnología con la que se implementará el proyecto, una remodelación de los paquetes dando lugar a los subsistemas de diseño, el desarrollo de los casos de uso a nivel de diseño y la propuesta de arquitectura de diseño.

### 3.1. Tecnologías elegidas

En el “Anexo II – Especificación de Requisitos” en la propuesta de arquitectura se vieron dos paquetes claramente diferenciados, Gestión de Prendas, que llevaría todo lo relacionado desde la presentación de las prendas a la obtención de las mismas y Gestión de Usuario que lleva desde la interacción con el usuario hasta iniciar sesión y registrarse.

Para la parte de prendas, que se obtendrán mediante *web scraping*, se ha elegido el *framework* de Python llamado Scrapy. Para la parte de usuario y toda la interacción se ha elegido el *framework* de Python llamado Flask, que está especializado en la creación de aplicaciones web. El funcionamiento de ambos *frameworks* está explicado en la “Memoria Principal”. Además, dado que las prendas y los usuarios deben de ser almacenados de forma persistente se utilizará el gestor de base de datos MariaDB para el almacenamiento de los mismos.

La elección de las tecnologías es clave a la hora de la organizar las clases y los subsistemas ya que cada *framework* tiene un modo único de funcionamiento con una estructura propia que afecta a dicha organización.

### 3.2. Subsistemas de diseño

Se mostrarán los distintos subsistemas que componen el proyecto y que suponen una evolución de los paquetes de análisis. El paquete Gestión de Prendas se ha renombrado como *prendas* y el de Gestión de Usuarios como *galería*. El paquete de *galería\_gestion* se corresponde con clases de control y modelo. La Galería será el nombre que se le dará a la aplicación web.

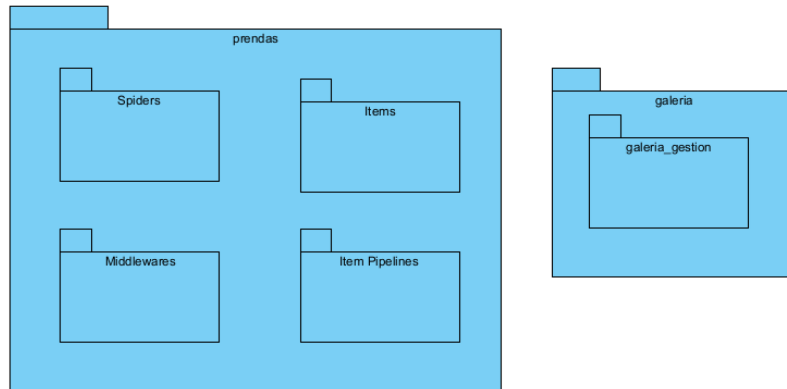


Ilustración 1 - Subsistemas de diseño

### 3.3. Diagrama de clases de diseño

En este apartado se muestran las clases que hay dentro de los distintos subsistemas. La principal diferencia entre las clases a nivel de análisis y a nivel de diseño es que los atributos están más desarrollados, hay métodos y se han refinado las relaciones existentes entre ellas.

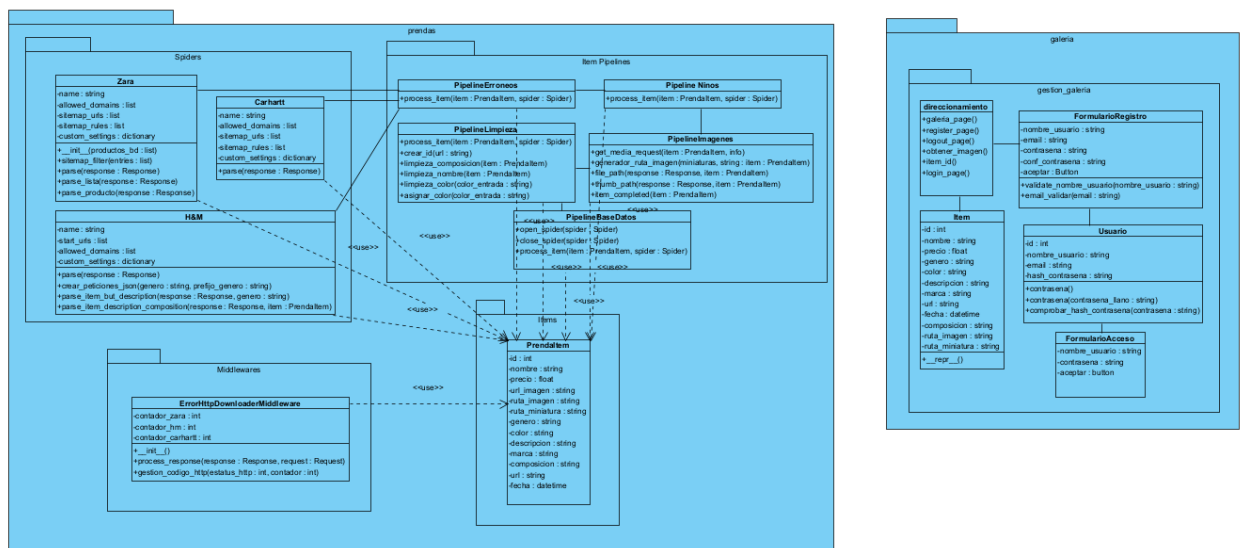


Ilustración 2 - Diagrama de clases de diseño

### 3.4. Casos de uso de diseño

Los casos de uso de diseño, a diferencia que en análisis y obtención de requisitos, se les pueden dar distintos puntos de vista expresados con distintos tipos de diagramas. Los diagramas utilizados para definir los casos de uso de diseño son los diagramas de estado y los diagramas de secuencia de diseño.

- CU – 001 Ver todas las prendas

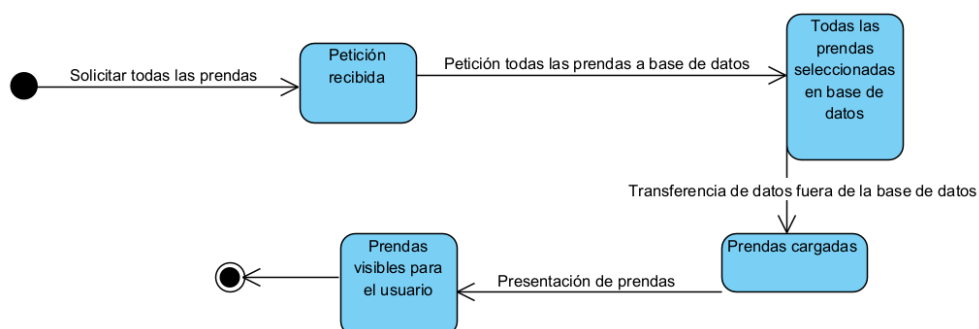


Ilustración 3 - Diagrama de estados de CU - 001

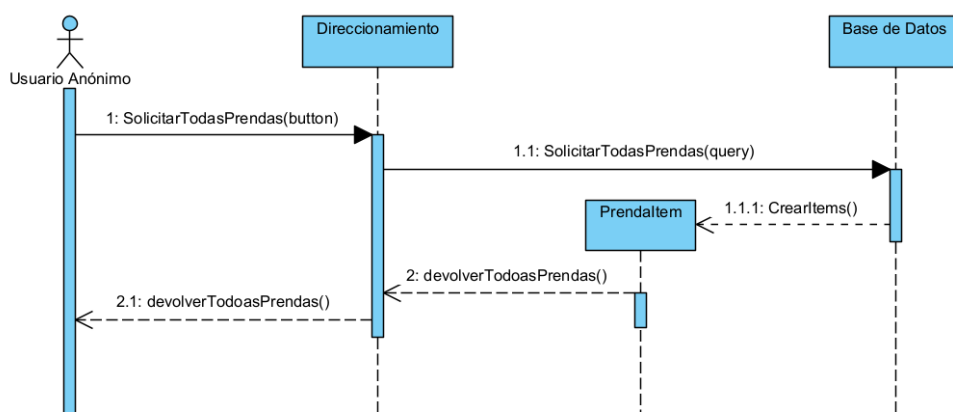


Ilustración 4 - Diagrama de secuencia de diseño de CU - 001

- CU – 002 Filtrar prendas

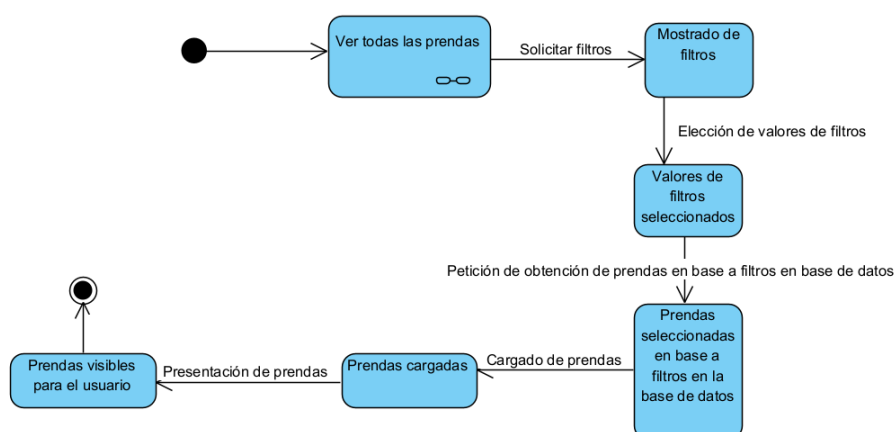


Ilustración 5 - Diagrama de estados de CU - 002

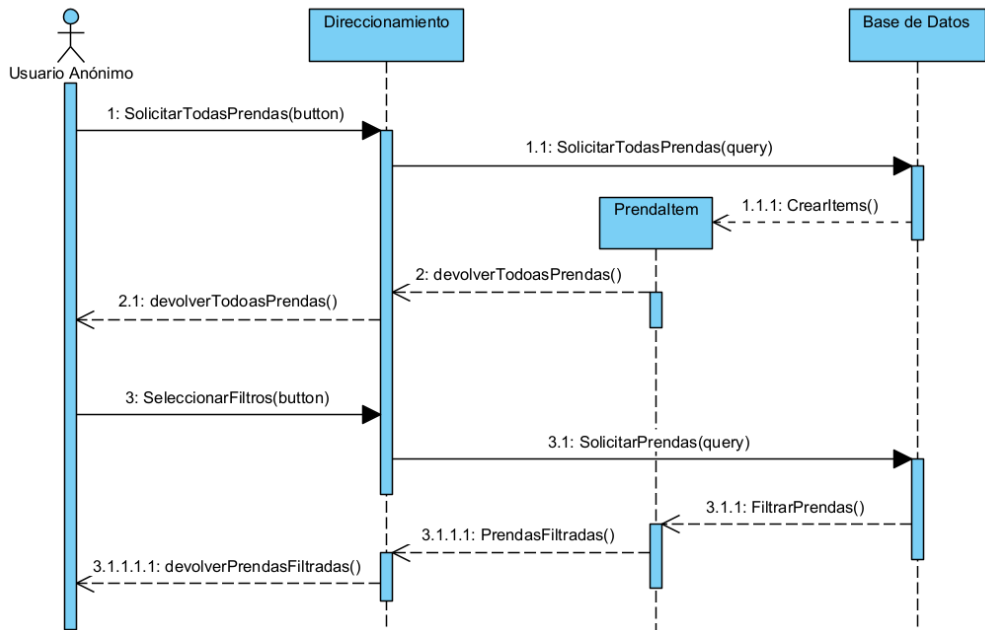


Ilustración 6 - Diagrama de secuencia de diseño de CU - 002

- CU – 003 Ver Prenda

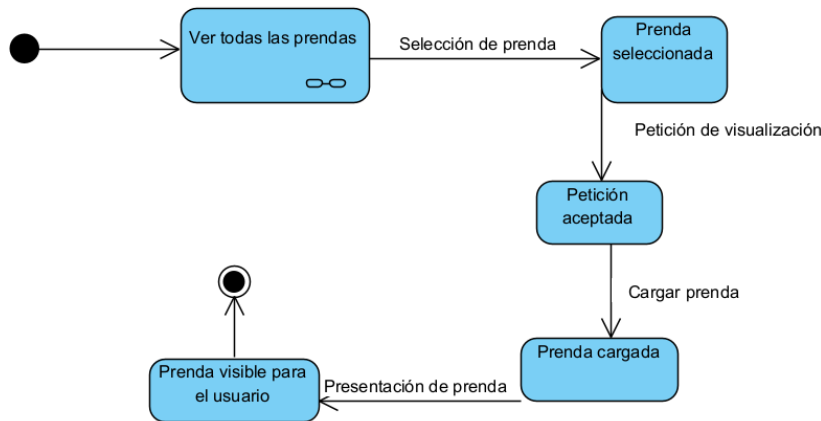


Ilustración 7 - Diagrama de estados de CU - 003

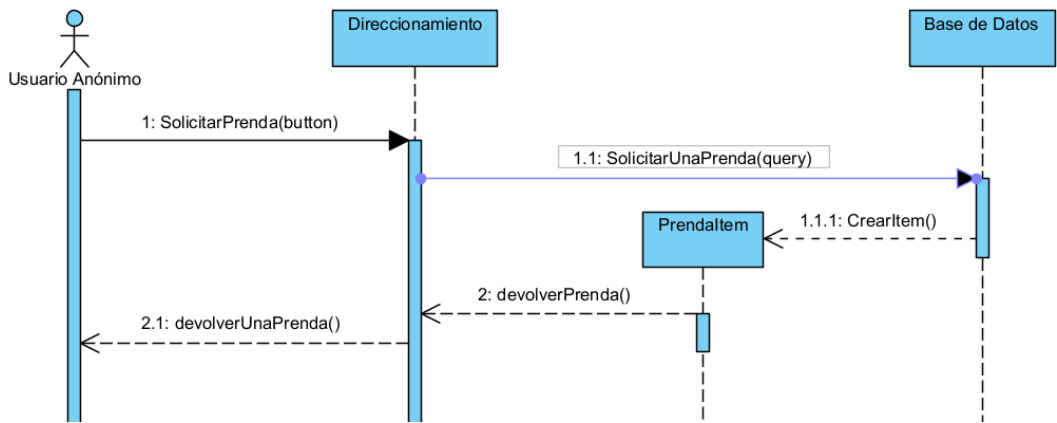


Ilustración 8 - Diagrama de secuencia de diseño de CU - 003

- CU – 004 Obtener catálogo

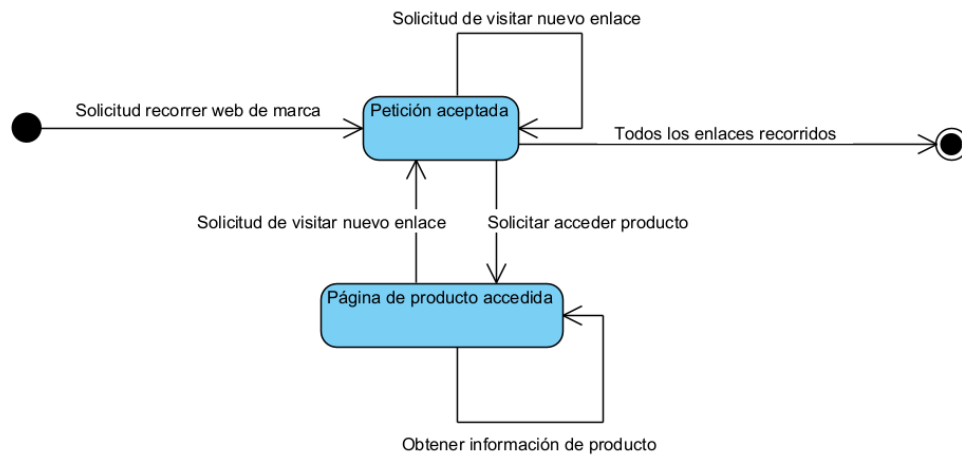


Ilustración 9 - Diagrama de estados de CU - 004

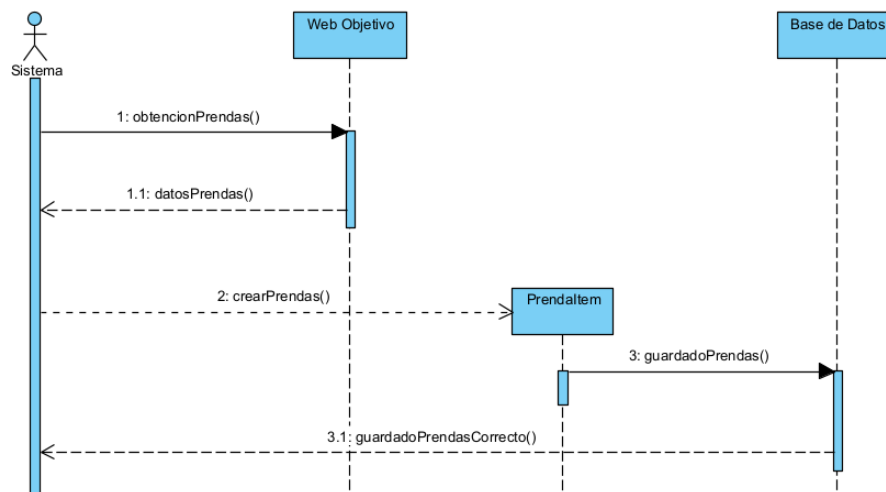


Ilustración 10 - Diagrama de secuencia de diseño de CU - 004

- CU – 005 Iniciar sesión

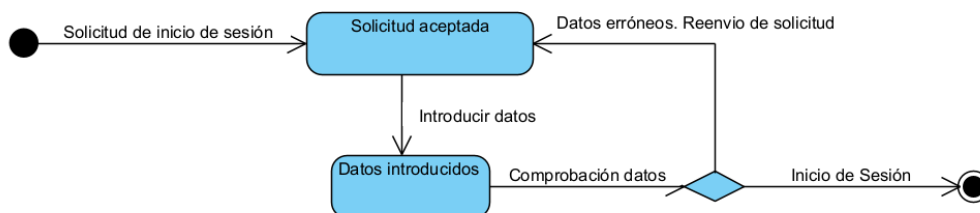


Ilustración 11 - Diagrama de estados de CU - 005

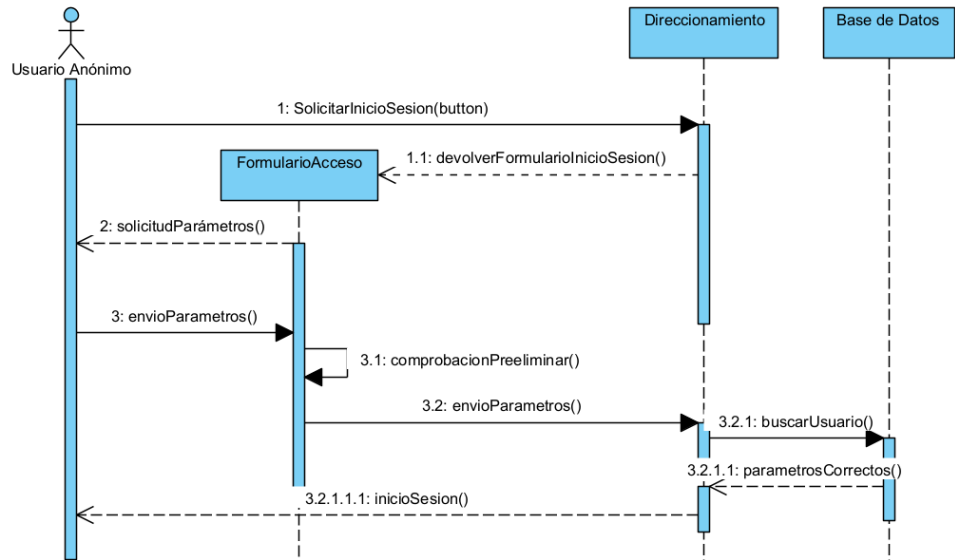


Ilustración 12 - Diagrama de secuencia de diseño de CU – 005

- CU – 006 Crear Cuenta de Usuario

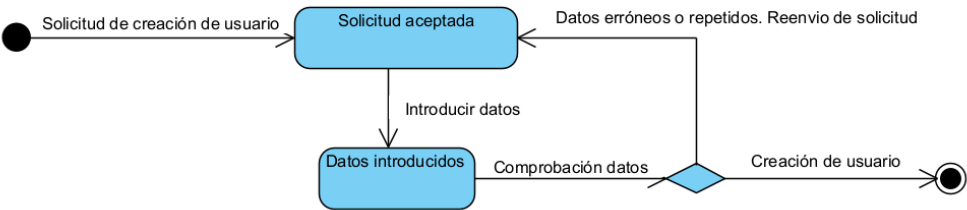


Ilustración 13 - Diagrama de estados de CU - 006

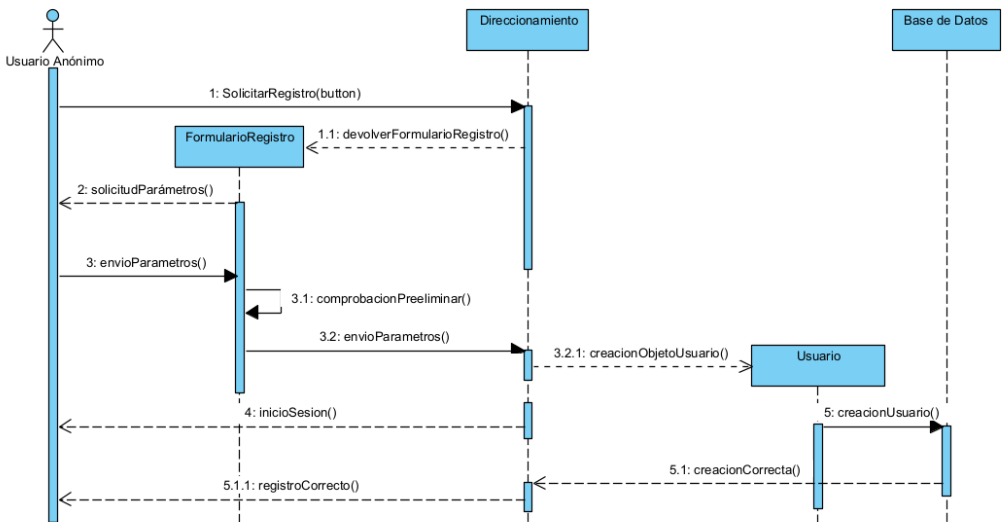


Ilustración 14 - Diagrama de secuencia de diseño de CU - 006



### 3.5. Diseño de la base de datos

Las únicas clases cuyos objetos se necesitan estén almacenadas de forma persistente son las clases de *PrendaItem* y *Usuario* para guardar todas las prendas que se obtengan y los usuarios que se registren.

El gestor de base de datos elegido es MariaDB.

Además, como se ve en el punto de “3.3 Diagrama de clases de diseño” no tienen ninguna relación entre ellas por lo que se creará una base de datos llamada *prendas* y dos tablas separadas sin relación.

Primero se explicará el diseño de la tabla de *item*, cuya finalidad es guardar los objetos *PrendaItem*. Las distintas columnas, en este caso campos, son los especificados en la clase *PrendaItem*. No se admiten valores nulos y el identificador será clave primaria.

Field	Type	Null	Key	Default	Extra
id	bigint(20)	NO	PRI	NULL	
nombre	varchar(100)	NO		NULL	
precio	float	NO		NULL	
url_imagen	varchar(250)	NO		NULL	
genero	varchar(30)	NO		NULL	
color	varchar(30)	NO		NULL	
descripcion	varchar(800)	NO		NULL	
marca	varchar(10)	NO		NULL	
url	varchar(200)	NO		NULL	
fecha	datetime	NO		NULL	
composicion	varchar(100)	NO		NULL	
ruta_imagen	varchar(100)	NO		NULL	
ruta_miniativa	varchar(100)	NO		NULL	

Ilustración 15 - Diseño de datos de la tabla item

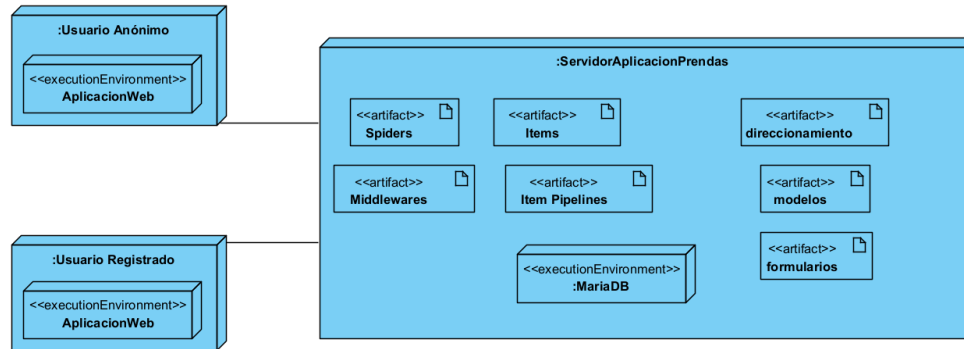
La segunda tabla será para almacenar los usuarios y se llama *usuario*. Las distintas columnas son los atributos especificados en los campos de la clase *Usuario*. No se admiten valores nulos, el campo *id* es la clave primaria y *nombre\_usuario* y *email* deben ser valores únicos para cada usuario, es decir, dos usuarios distintos no pueden tener ni *email* ni *nombre\_usuario* parecido.

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
nombre_usuario	varchar(15)	NO	UNI	NULL	
email	varchar(50)	NO	UNI	NULL	
hash_contraseña	varchar(64)	NO		NULL	

Ilustración 16 - Diseño de datos de la tabla usuario

### 3.6. Modelo de despliegue

El modelo de despliegue representa cómo se distribuirá el sistema de forma física, tanto las distintas máquinas que intervienen como los artefactos que indican cómo se distribuirá el código.



*Ilustración 17 - Diagrama de despliegue*

## 4. IMPLEMENTACIÓN

La fase de implementación es la última etapa de desarrollo del producto en la Ingeniería del Software. En esta etapa se realiza la última etapa de refinamiento y se implementa en los lenguajes de programación elegidos.

El elemento que más importancia cobra en esta etapa es el componente, que se corresponde con el fichero que contiene las clases definidas en la etapa de diseño. Los puntos que se describirán será la definición de los subsistemas con los componentes y la asignación de las clases de diseño a los componentes.

### 4.1. Subsistemas de implementación con componentes

Se asignarán los distintos componentes a los subsistemas y sus distintas relaciones.

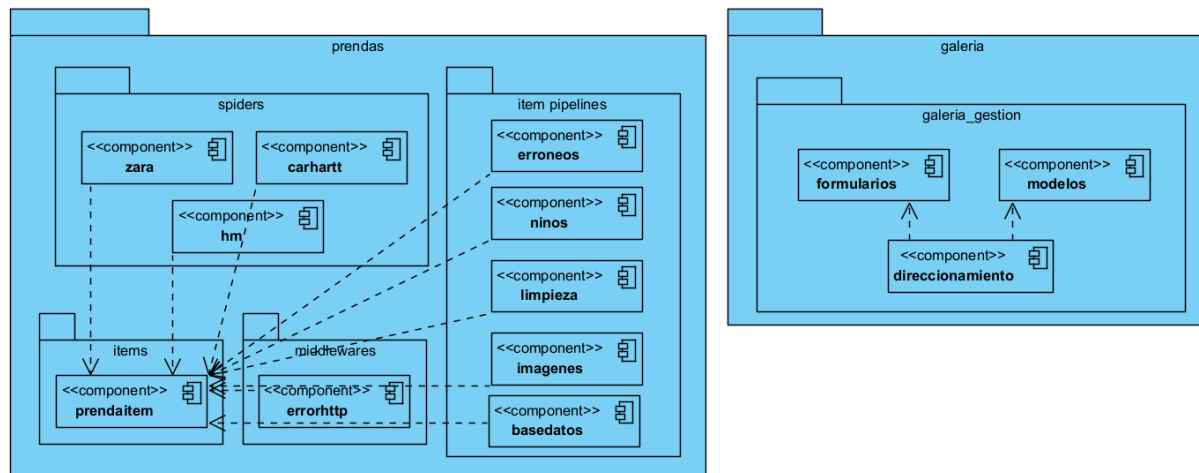


Ilustración 18 - Diagrama de subsistemas con componentes de implementación

## 4.2. Clases incluidas en cada componente

Se mostrarán qué clases pertenecen a cada componente.

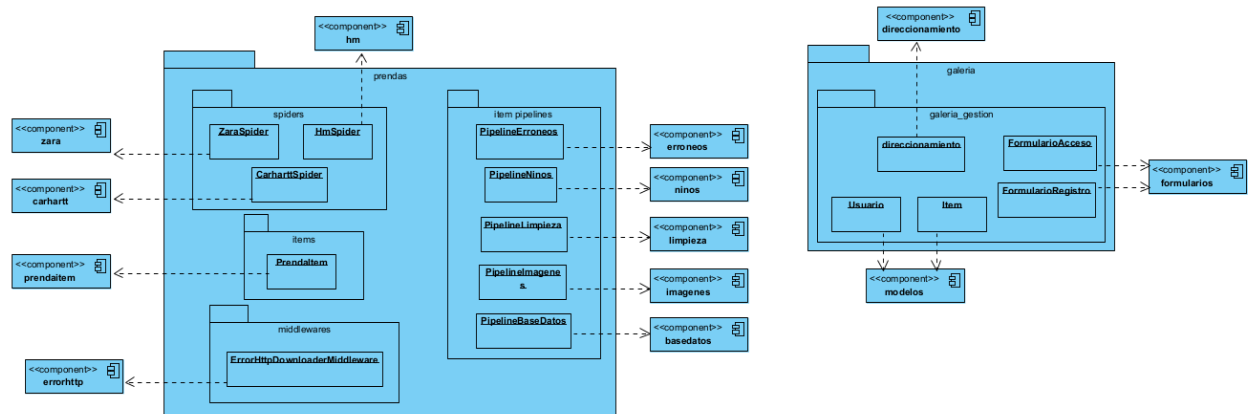


Ilustración 19 - Clases por cada componente