

## Escuela de Ingeniería Industrial

### TRABAJO FIN DE GRADO

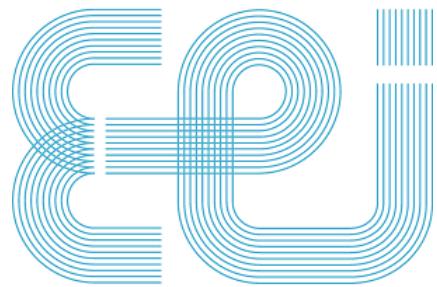
*Diseño de una aplicación informática para la gestión de un almacén sin automatización con distintas interfaces para los operarios*

**Grado en Ingeniería en Electrónica Industrial y Automática**

**ALUMNO:** José Tomás Torre Pedroarena

**DIRECTORES:** Joaquín López Fernández

UniversidadeVigo



## Escuela de Ingeniería Industrial

### **TRABAJO FIN DE GRADO**

*Diseño de una aplicación informática para la gestión de un almacén sin automatización con distintas interfaces para los operarios*

**Grado en Ingeniería en Electrónica Industrial y Automática**

**Documento**

**MEMORIA**

UniversidadeVigo

## ÍNDICE

Índice de figuras .....	3
Índice de tablas .....	4
1 Introducción .....	6
1.1 Contexto y motivación del proyecto .....	6
1.2 Problema a resolver .....	7
1.3 Objetivos del TFG .....	7
2 Estado del arte .....	9
2.1 Sistemas de gestión de almacenes (WMS) .....	9
2.2 Funcionalidades habituales en aplicaciones de gestión de almacenes .....	9
2.3 Aplicaciones comerciales de gestión de almacenes .....	10
2.4 Justificación de la solución propuesta .....	10
3 Marco teórico y tecnológico .....	11
3.1 Aplicaciones multiplataforma .....	11
3.2 Arquitecturas cliente-servidor .....	11
3.3 Aplicaciones móviles Android .....	12
3.4 Aplicaciones de escritorio .....	12
3.5 Comunicación entre aplicaciones (APIs REST) .....	13
3.6 Backend y framework Spring Boot .....	14
3.7 Acceso a la base de datos en la aplicación de escritorio .....	14
4 Análisis del sistema .....	15
4.1 Descripción general del sistema .....	15
4.2 Identificación de usuarios .....	15
4.3 Requisitos del sistema .....	16
4.3.1 Requisitos funcionales .....	16
4.3.2 Requisitos no funcionales .....	16
4.4 Restricciones técnicas .....	17
5 Diseño del sistema .....	18
5.1 Arquitectura general del sistema .....	18
5.2 Diseño de la aplicación Desktop .....	18
5.3 Diseño de la aplicación Android .....	19
5.4 Diseño de la base de datos .....	20
5.5 Diseño de la API .....	20
5.6 Diagramas UML .....	20
5.6.1 Diagrama de clases .....	20
5.6.2 Diagrama de secuencia .....	20
5.6.3 Diagrama de despliegue .....	20
5.7 Diseño de la interfaz de usuario .....	20
6 Implementación .....	22
6.1 Herramientas y tecnologías utilizadas .....	22
6.2 Implementación de la aplicación Desktop .....	22
6.3 Implementación de la aplicación Android .....	22

**DISEÑO DE UNA APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE UN ALMACÉN SIN AUTOMATIZACIÓN CON DISTINTAS INTERFACES PARA LOS OPERARIOS**

**JOSÉ TOMÁS TORRE PEDROARENA**

---

6.4 Implementación del backend / servidor.....	22
6.5 Implementación de la base de datos .....	22
6.6 Seguridad y gestión de errores.....	22
6.7 Control de versiones y gestión del proyecto .....	22
7 Resultados y discusión.....	23
7.1 Resultados del desarrollo.....	23
7.2 Evaluación del cumplimiento de objetivos .....	23
7.3 Ventajas de la solución propuesta .....	23
7.4 Limitaciones detectadas .....	23
8 Conclusiones y trabajos futuros .....	24
8.1 Conclusiones .....	24
8.2 Líneas de mejora y trabajos futuros .....	24

## ÍNDICE DE FIGURAS

1 Almacén de consolidación en entorno industrial .....	6
2 Arquitectura cliente-servidor .....	11
3 Comunicación mediante APIs REST .....	13
4 Comunicación directa entre la aplicación de escritorio y la base de datos .....	14
5 Arquitectura general del sistema cliente-servidor.....	18
6 Diagrama de navegación entre ventanas de la aplicación Desktop.....	19
7 Ejemplo de navegación entre ventanas de la aplicación Desktop (desde <i>Login</i> hasta <i>Orden Compras</i> ).....	19

## ÍNDICE DE TABLAS

1 Permisos de acceso a funcionalidades del sistema según rol de usuario. ....	15
---	----

## GLOSARIO DE SIGLAS

**API** Interfaz de Programación de Aplicaciones.

**ERP** Planificación de Recursos Empresariales.

**GUI** Interfaz Gráfica de Usuario.

**HTTP** Hypertext Transfer Protocol.

**HTTPS** Hypertext Transfer Protocol Secure.

**JSON** JavaScript Object Notation.

**REST** Transferencia de Estado Representacional.

**SQL** Lenguaje de Consulta Estructurado.

**WMS** Sistema de Gestión de Almacenes.

**XML** Extensible Markup Language.

## DEFINICIONES

**Backend** Conjunto de servidores y componentes software encargados de proporcionar servicios, gestionar la lógica de negocio y permitir el acceso a recursos y datos desde aplicaciones cliente.

**Framework** Estructura de desarrollo compuesta por componentes de software reutilizables, librerías y convenciones que facilitan la creación de aplicaciones de forma estandarizada y eficiente.

**JDBC** Java Database Connectivity, API estándar de Java que permite a las aplicaciones acceder y gestionar bases de datos relacionales mediante la ejecución de consultas SQL y la manipulación de los resultados obtenidos.

**ResultSet** Estructura de datos utilizada en JDBC para representar el conjunto de resultados devuelto por una consulta a una base de datos, permitiendo recorrer fila a fila los registros obtenidos y acceder a los valores de cada columna.

**SKU** Stock Keeping Unit, identificador único de un producto en el inventario.

## 1 INTRODUCCIÓN

La optimización de los procesos industriales y logísticos constituye uno de los pilares fundamentales en los entornos productivos actuales. Aspectos como el control operativo, la trazabilidad de los materiales o la reducción de errores influyen de manera directa tanto en los costes como en la calidad del servicio ofrecido. Dentro de este escenario, el almacén adquiere un papel determinante en la cadena de suministro, al actuar como punto de conexión entre producción, transporte y distribución.

Entre los distintos modelos de almacenamiento existentes, los almacenes de consolidación desempeñan una función específica centrada en la agrupación, organización y preparación de mercancía procedente de múltiples orígenes para su expedición conjunta. Este tipo de instalaciones presenta retos operativos propios, entre los que destacan un elevado número de movimientos, una rotación continua de productos y la necesidad de mantener un seguimiento preciso de palets, ubicaciones y pedidos.

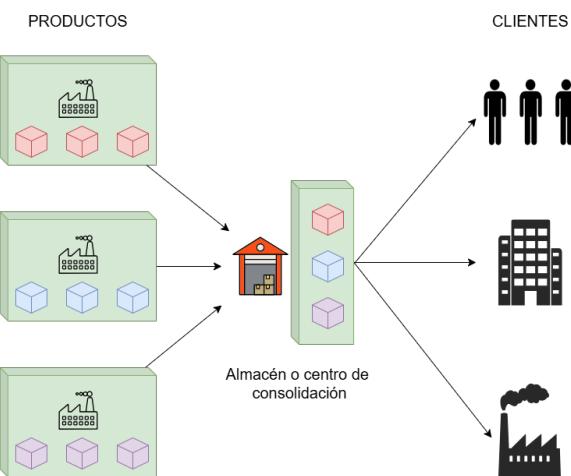


Figura 1: Almacén de consolidación en entorno industrial

La incorporación de sistemas digitales en la gestión de estos almacenes permite un mayor control del inventario, una localización más precisa de los materiales y una coordinación más ordenada de los flujos de entrada y salida. En este contexto se enmarca el presente Trabajo Fin de Grado, cuyo propósito es el desarrollo de una solución técnica orientada de forma específica a la gestión de almacenes de consolidación, concebida como apoyo directo a la operativa industrial diaria.

El proyecto aborda el diseño e implementación de un sistema compuesto por una aplicación de escritorio, una aplicación móvil para dispositivos Android y un backend responsable del tratamiento y almacenamiento de la información. El planteamiento adoptado responde a un enfoque práctico, alineado con los conocimientos adquiridos en la titulación de Ingeniería en Electrónica Industrial y Automática, y orientado a la aplicación real de tecnologías software como soporte a procesos logísticos.

### 1.1 Contexto y motivación del proyecto

En entornos industriales y logísticos, la gestión del almacén condiciona de forma directa la continuidad de la producción y el cumplimiento de los plazos establecidos. En los almacenes de consolidación, una gestión deficiente puede derivar en desajustes de inventario, tiempos improductivos durante la preparación de pedidos, errores en la agrupación de mercancía o dificultades para conocer el estado real de los materiales.

Existen soluciones comerciales destinadas a la gestión de almacenes, aunque muchas de ellas están orientadas a grandes centros logísticos generalistas o incorporan un nivel de complejidad elevado. Esta situación complica su implantación en instalaciones de menor escala o con flujos de trabajo muy concretos, donde la inversión en infraestructura, formación o consultoría resulta poco viable.

La motivación principal de este proyecto surge de la necesidad de contar con una herramienta accesible y comprensible, pensada para entornos que carecen de soluciones específicas para la gestión de almacenes de consolidación. La propuesta se centra en cubrir las funciones esenciales de este tipo de instalaciones e integrarse de forma natural en la rutina diaria del almacén mediante interfaces claras y orientadas al usuario. La combinación de una aplicación de escritorio para tareas de planificación y una aplicación móvil para la operativa en planta responde a situaciones habituales en entornos industriales reales.

Desde el punto de vista académico, el desarrollo del proyecto permite aplicar conocimientos relacionados con sistemas de información, arquitectura de software, gestión de datos y diseño de interfaces, reforzando la formación técnica en automatización y electrónica industrial con una visión orientada a la resolución de problemas logísticos.

## **1.2 Problema a resolver**

El problema abordado en este Trabajo Fin de Grado se centra en el desarrollo de sistemas integrados y adaptados a la realidad de los almacenes de consolidación en entornos industriales. En concreto, se plantea la necesidad de gestionar de manera estructurada la información asociada a productos, palets, ubicaciones, movimientos y pedidos.

En numerosas instalaciones, estas tareas se apoyan en herramientas genéricas, soluciones parciales o procedimientos manuales, lo que incrementa el riesgo de errores y dificulta el acceso a información actualizada y coherente. Este escenario puede traducirse en pérdidas de material, desajustes de inventario y una trazabilidad limitada, especialmente en instalaciones con alta rotación de mercancía.

El sistema propuesto plantea una respuesta a esta problemática mediante una solución que:

- Centraliza la información del almacén de consolidación en una base de datos común.
- Permite el control estructurado del inventario y la gestión de ubicaciones.
- Apoya la gestión de pedidos y el registro de movimientos durante los procesos de consolidación.

El alcance del proyecto se limita al desarrollo de un sistema de apoyo a la gestión del almacén, sin intervenir en el control directo de equipos automatizados. Aun así, la arquitectura planteada contempla la posibilidad de integración futura con sistemas de control de planta o automatización industrial.

## **1.3 Objetivos del TFG**

El objetivo general de este Trabajo Fin de Grado consiste en el desarrollo de un sistema de gestión orientado a almacenes de consolidación en entornos industriales, con el fin de mejorar el control, la organización y la trazabilidad de los materiales mediante aplicaciones software integradas.

A partir de este objetivo general, se definen los siguientes objetivos específicos:

- Estudiar las necesidades operativas propias de un almacén de consolidación industrial.
- Definir una arquitectura que permita la comunicación entre aplicaciones de escritorio, móviles y un backend común.

**DISEÑO DE UNA APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE UN ALMACÉN SIN AUTOMATIZACIÓN CON DISTINTAS INTERFACES PARA LOS OPERARIOS**

**JOSÉ TOMÁS TORRE PEDROARENA**

---

- Desarrollar la gestión de productos, palets, ubicaciones y pedidos.
- Implementar un sistema de almacenamiento de datos que asegure la coherencia y disponibilidad de la información.
- Diseñar interfaces orientadas a la operativa diaria del personal del almacén.

## 2 ESTADO DEL ARTE

En este capítulo se presenta el estado del arte relacionado con los sistemas de gestión de almacenes, con especial atención a los almacenes de consolidación. Se describen los conceptos generales asociados a los Sistemas de Gestión de Almacenes (WMSs), las funcionalidades habituales que incorporan este tipo de aplicaciones y algunas de las soluciones comerciales existentes, con el objetivo de contextualizar la solución propuesta en este Trabajo Fin de Grado.

### 2.1 Sistemas de gestión de almacenes (WMS)

Un Sistema de Gestión de Almacenes, conocido como WMS (*Warehouse Management System*), es una herramienta software diseñada para apoyar y optimizar las operaciones que se llevan a cabo en un almacén. Su principal función es gestionar de forma estructurada la información relacionada con el inventario, las ubicaciones, los movimientos de mercancía y los procesos de entrada y salida de materiales.

Los sistemas WMS actúan como un elemento central dentro de la cadena logística, proporcionando visibilidad sobre el estado del almacén y facilitando la toma de decisiones operativas. En entornos industriales, estos sistemas permiten reducir errores asociados a la gestión manual, mejorar la trazabilidad de los productos y optimizar el uso del espacio y de los recursos disponibles.

En función del tipo de instalación, los WMS pueden adaptarse a diferentes escenarios, como almacenes de producción, centros de distribución o almacenes de consolidación. En este último caso, el sistema debe gestionar un elevado volumen de movimientos, coordinar la agrupación de mercancía procedente de diferentes orígenes y garantizar una correcta trazabilidad de palets y pedidos hasta su expedición.

Desde un punto de vista funcional, un WMS suele estructurarse en distintos módulos que cubren aspectos como la gestión de inventario, el control de ubicaciones, la planificación de operaciones y la generación de informes. Asimismo, estos sistemas pueden integrarse con otros sistemas de información, como sistemas de planificación de recursos empresariales (ERP) o sistemas de control industrial, dependiendo del nivel de automatización de la instalación.

### 2.2 Funcionalidades habituales en aplicaciones de gestión de almacenes

Las aplicaciones de gestión de almacenes incorporan un conjunto de funcionalidades orientadas a cubrir las necesidades operativas básicas de una instalación logística o industrial. Aunque estas funcionalidades pueden variar en función del tipo de almacén y del sector, existen una serie de características comunes ampliamente presentes en los sistemas WMS.

Entre las funcionalidades más habituales se encuentran la gestión de la recepción de mercancía, que permite registrar la entrada de productos en el almacén y asociarlos a ubicaciones específicas. Asimismo, la gestión de ubicaciones resulta fundamental para organizar el espacio disponible y facilitar la localización de materiales durante las operaciones diarias.

Otra funcionalidad clave es la gestión del inventario, que proporciona información actualizada sobre las existencias disponibles y permite detectar desajustes o incidencias. En el caso de los almacenes de consolidación, esta funcionalidad adquiere especial relevancia debido a la alta rotación de productos y a la necesidad de coordinar múltiples movimientos de entrada y salida.

Las aplicaciones WMS también suelen incluir herramientas para la gestión de pedidos y la preparación de expediciones, facilitando la agrupación de mercancía y el seguimiento de los envíos. La trazabilidad de los materiales, especialmente a nivel de palet, constituye otro aspecto esencial, ya que permite conocer el historial de movimientos y garantizar un control adecuado de

la mercancía.

Finalmente, muchas aplicaciones incorporan funcionalidades de apoyo como la generación de informes, el control de usuarios y permisos, y la integración con dispositivos móviles, lo que resulta especialmente útil para la operativa en planta y para la reducción de errores en tareas repetitivas.

## 2.3 Aplicaciones comerciales de gestión de almacenes

En el mercado existen numerosas soluciones comerciales orientadas a la gestión de almacenes, que ofrecen un amplio abanico de funcionalidades y niveles de complejidad. Estas soluciones suelen formar parte de plataformas logísticas o sistemas empresariales más amplios, y están diseñadas para adaptarse a diferentes tipos de instalaciones y volúmenes de operación.

Algunas de estas aplicaciones están orientadas a grandes centros logísticos o a entornos altamente automatizados, incorporando funcionalidades avanzadas de optimización y control. Sin embargo, este enfoque puede suponer una barrera para su adopción en instalaciones de menor escala o en almacenes de consolidación con procesos específicos, debido a la complejidad de configuración y a los costes asociados a su implantación y mantenimiento.

Otras soluciones comerciales ofrecen versiones más simplificadas, pero aun así suelen estar basadas en flujos de trabajo genéricos que no siempre se ajustan de forma óptima a las particularidades de cada instalación. En muchos casos, la adaptación a procesos concretos requiere desarrollos adicionales o modificaciones en la operativa existente, lo que reduce la flexibilidad del sistema.

En este contexto, se observa una brecha entre las soluciones altamente especializadas y complejas, y la necesidad de herramientas más accesibles y adaptables a entornos industriales concretos, como los almacenes de consolidación que no cuentan con sistemas de gestión avanzados.

## 2.4 Justificación de la solución propuesta

A partir del análisis del estado del arte, se pone de manifiesto la necesidad de disponer de soluciones de gestión de almacenes que, sin alcanzar la complejidad de los grandes sistemas comerciales, permitan cubrir de forma eficaz las necesidades operativas de instalaciones concretas, como los almacenes de consolidación.

La solución propuesta en este Trabajo Fin de Grado se justifica por su enfoque específico hacia este tipo de almacenes, priorizando la simplicidad de uso, la adaptación a flujos de trabajo reales y la integración de distintas plataformas de acceso. El sistema se concibe como una herramienta de apoyo a la operativa industrial, facilitando la gestión de inventario, la trazabilidad de palets y la coordinación de pedidos sin imponer una reestructuración completa de los procesos existentes.

Además, la combinación de una aplicación de escritorio para tareas de gestión y planificación, junto con una aplicación móvil para la operativa en planta, permite cubrir distintos escenarios de uso habituales en entornos industriales. Este enfoque contribuye a mejorar la accesibilidad a la información y a reducir errores derivados de la introducción manual de datos.

En conjunto, la propuesta se sitúa como una alternativa intermedia entre soluciones comerciales complejas y la ausencia de herramientas específicas, ofreciendo una base flexible que puede evolucionar e integrarse en el futuro con otros sistemas de automatización industrial.

### 3 MARCO TEÓRICO Y TECNOLÓGICO

En este capítulo se presentan los fundamentos teóricos y tecnológicos que se emplearon en el desarrollo del sistema propuesto. Se abordan los conceptos relacionados con los sistemas de gestión de almacenes, las arquitecturas cliente-servidor, el desarrollo de aplicaciones multiplataforma, las aplicaciones móviles Android, las aplicaciones de escritorio y los mecanismos de comunicación entre aplicaciones, incluyendo el uso de Interfaz de Programación de Aplicaciones (APIs) de tipo Transferencia de Estado Representacional (REST) en determinados componentes del sistema.

#### 3.1 Aplicaciones multiplataforma

¿Cómo logramos que el jefe de almacén y el operario empleen el mismo sistema e interactúen con los mismos datos? La respuesta está en el desarrollo multiplataforma. En un entorno donde conviven ordenadores de oficina y terminales móviles, es vital que la lógica del sistema no se fragmente. Al compartir una base común (base de datos) ambos dispositivos deben interactuar entre sí y con el servidor de manera coherente.

Desde un punto de vista funcional, la adopción de soluciones multiplataforma facilita la reutilización de la lógica de negocio y la centralización de la información, reduciendo los costes de mantenimiento y mejorando la coherencia del sistema. En el ámbito de la gestión de almacenes, este enfoque permite que distintos perfiles de usuario accedan al sistema desde interfaces adaptadas a sus necesidades manteniendo una única fuente de datos consistente, por ejemplo: un operario necesita un dispositivo móvil para poder realizar sus actividades, por tanto, usará un dispositivo móvil mientras que el jefe de almacén, que requiere una interfaz más completa, accederá desde un ordenador de oficina.

No obstante, el desarrollo multiplataforma también plantea retos, como la adaptación de la interfaz de usuario a diferentes dispositivos o la necesidad de garantizar un comportamiento homogéneo, es decir, que las operaciones realizadas desde un dispositivo se reflejen correctamente en el otro. Estos aspectos deben ser considerados cuidadosamente durante el diseño y la implementación del sistema.

#### 3.2 Arquitecturas cliente-servidor

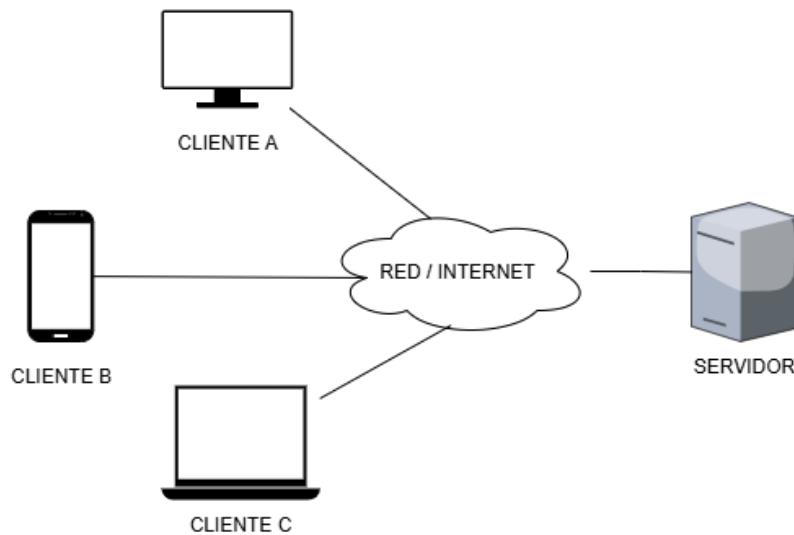


Figura 2: Arquitectura cliente-servidor

La arquitectura cliente-servidor es un modelo ampliamente utilizado en sistemas distribuidos<sup>1</sup>, en el que las funciones del sistema se reparten entre componentes cliente y un servidor central. Los clientes se encargan de la interacción con el usuario, mientras que el servidor gestiona la lógica de negocio y la persistencia de los datos.

Este modelo resulta adecuado y conveniente para sistemas de gestión en entornos industriales, dado que permite centralizar la información y garantizar el acceso concurrente desde múltiples dispositivos. El sistema desarrollado se despliega en una red interna privada, prescindiendo del acceso a Internet, donde tanto la aplicación de escritorio como las dispositivos móviles Android funcionan como clientes que se comunican con el servidor central.

Además, este tipo de arquitectura favorece la escalabilidad del sistema y su posible integración futura con otros sistemas, como plataformas empresariales o sistemas de automatización industrial, al proporcionar un punto centralizado de acceso a los servicios y a los datos.

### 3.3 Aplicaciones móviles Android

Android es uno de los sistemas operativos más extendidos en dispositivos móviles, “un 78,8 % de los españoles con smartphone tuvieron Android como sistema operativo”<sup>2</sup>.

Según la documentación oficial de Android este es una arquitectura basada en componentes como actividades, servicios, entre otros, que permiten gestionar la interfaz de usuario y el ciclo de vida de la aplicación [3].

En entornos industriales y logísticos, el uso de aplicaciones móviles facilita el acceso a la información en tiempo real y reduce la necesidad de desplazamientos innecesarios o registros manuales. En particular, en almacenes de consolidación, las aplicaciones Android permiten registrar movimientos, consultar inventario o identificar palets directamente en el punto de operación.

Dentro del sistema desarrollado, la aplicación android pretende ser una herramienta de apoyo para los operarios del almacén, proporcionando una interfaz sencilla para el registro de las operaciones que estos realizan en planta.

### 3.4 Aplicaciones de escritorio

Las aplicaciones de escritorio continúan desempeñando un papel fundamental en entornos industriales para tareas de gestión, supervisión y planificación. Estas aplicaciones suelen ejecutarse en estaciones de trabajo fijas y permiten ofrecer interfaces más completas para la visualización y administración de grandes volúmenes de información.

En el sistema desarrollado, la aplicación de escritorio se concibe como la herramienta principal para la gestión del almacén de consolidación, permitiendo la administración de productos, ubicaciones, pedidos y usuarios. Este tipo de aplicación resulta especialmente adecuada para tareas que requieren una visión global del sistema y un mayor nivel de detalle en la información presentada.

Asimismo, la aplicación integra todas las funcionalidades de gestión y administración y está destinada principalmente a usuarios con privilegios elevados (SysAdmin, Gestor Almacén, Administración). La gestión de roles y permisos se detalla en la sección 4.2. La aplicación de escritorio también puede ser usada por operarios en caso de ser necesario.

<sup>1</sup>Universidade de Vigo. “Arquitectura Cliente/Servidor.” Material docente de la asignatura Sistemas Cliente/Servidor. Último acceso: enero 2026. dirección: <https://ccia.esei.uvigo.es/docencia/SCS/1011/transparencias/Tema1.pdf> p. 1

<sup>2</sup>Comisión Nacional de los Mercados y la Competencia. “Panel de hogares: usos de Internet.” Gráfica “Sistema operativo del smartphone”. Último acceso: enero 2026. dirección: <https://www.cnmc.es/prensa/panel-hogares-usos-internet-20231103> Fragmento: gráfica “Sistema operativo del smartphone”

La combinación de aplicaciones de escritorio y móviles permite adaptar la interacción con el sistema a distintos contextos de trabajo, teniendo en cuenta que los perfiles de operario y gestor de almacén presentan necesidades diferenciadas. Esta aproximación contribuye a una experiencia de uso más ajustada y a un desarrollo más ordenado de la actividad dentro del almacén.

### 3.5 Comunicación entre aplicaciones (APIs REST)

La comunicación entre los distintos componentes del sistema puede realizarse mediante interfaces de programación de aplicaciones (APIs). Una API puede definirse como un “ [...] conjunto de comandos, funciones y protocolos informáticos que permiten crear programas que interactúen con otras aplicaciones [...]”<sup>3</sup>. Este mecanismo resulta especialmente adecuado en escenarios donde existen clientes heterogéneos que requieren acceso controlado a la información del sistema.

En este contexto, las APIs de tipo REST constituyen uno de los enfoques más utilizados para la comunicación entre sistemas distribuidos. REST es un conjunto de principios arquitectónicos orientados al diseño de interfaces entre sistemas, basado en el uso del protocolo Hypertext Transfer Protocol (HTTP), o bien Hypertext Transfer Protocol Secure (HTTPS) como medio de comunicación, sin añadir capas adicionales de abstracción <sup>4</sup>.

Entre las principales características de las arquitecturas REST se encuentra el empleo de HTTP para la realización de operaciones sobre los recursos del sistema, así como el intercambio de datos mediante formatos estructurados, siendo JavaScript Object Notation (JSON) y Extensible Markup Language (XML) los más habituales. Los sistemas que implementan estos principios se denominan sistemas RESTful.

Una API REST, o API RESTful, es por tanto una interfaz que expone los recursos de un sistema siguiendo los principios REST, permitiendo que los clientes accedan a dichos recursos mediante peticiones HTTP estandarizadas y reciban las respuestas en formatos estructurados. Este enfoque favorece la interoperabilidad, el desacoplamiento entre clientes y servidor y la evolución independiente de los distintos componentes del sistema.

En entornos industriales y logísticos, el uso de APIs REST permite integrar aplicaciones de escritorio, aplicaciones móviles y sistemas externos, garantizando un acceso coherente y controlado a la información del sistema. Además, este tipo de interfaces facilita la escalabilidad de la solución y su posible integración futura con otros sistemas de información o plataformas de automatización industrial.

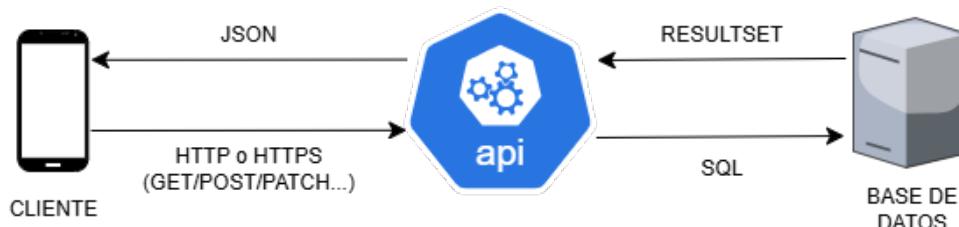


Figura 3: Comunicación mediante APIs REST

En el sistema desarrollado, el uso de una API REST se aplica específicamente en la comunicación entre la aplicación móvil Android y el backend del sistema, implementado mediante Spring Boot. La aplicación de escritorio, por su parte, se conecta directamente al servidor de base

<sup>3</sup>Asociación Española de Banca. “APIs.” Documento institucional. Último acceso: enero 2026. dirección: <https://s1.aebanca.es/wp-content/uploads/2018/05/apis.pdf> p. 1

<sup>4</sup>Iniciativa Aporta. “Buenas prácticas en el diseño de APIs y linked data.” Gobierno de España. dirección: [https://datos.gob.es/elearning/Unidades\\_Didacticas/Unidad\\_8/contenidos/descargas/unidad\\_imprimible.pdf](https://datos.gob.es/elearning/Unidades_Didacticas/Unidad_8/contenidos/descargas/unidad_imprimible.pdf) p. 10

de datos, actuando como un cliente especializado para tareas de gestión y administración. Esta aproximación permite adaptar el mecanismo de comunicación a las necesidades y características de cada tipo de cliente.

### 3.6 Backend y framework Spring Boot

Spring Boot es un Framework de desarrollo basado en el lenguaje Java que facilita la creación de aplicaciones de tipo Backend y servicios web. Está diseñado para simplificar la configuración y el despliegue de aplicaciones, proporcionando una estructura predefinida y mecanismos de autoconfiguración que reducen la necesidad de configuraciones manuales [6], [7], [8].

En el contexto de sistemas distribuidos, Spring Boot se utiliza habitualmente para implementar APIs de tipo REST, permitiendo exponer la lógica de negocio y el acceso a los datos mediante servicios accesibles por distintos clientes. Su integración con el ecosistema Java y su compatibilidad con tecnologías de persistencia lo convierten en una opción adecuada para el desarrollo de Backends en entornos industriales y empresariales.

### 3.7 Acceso a la base de datos en la aplicación de escritorio

En el sistema desarrollado se ha adoptado una estrategia diferenciada para el acceso a los datos en función del tipo de aplicación cliente. Mientras que la aplicación móvil Android accede a la información del sistema a través de una API de tipo REST, la aplicación de escritorio establece una conexión directa con el servidor de base de datos.

Esta decisión se fundamenta en el contexto de uso de la aplicación de escritorio, concebida como una herramienta de gestión interna destinada a tareas de administración, planificación y supervisión del almacén. Dicha aplicación se ejecuta en ordenadores de escritorio dentro de la red interna de la instalación y es utilizada por perfiles de usuario con permisos elevados (SysAdmin, Gestor Almacén, Administración), lo que permite asumir un entorno controlado desde el punto de vista de seguridad y acceso.

El acceso directo a la base de datos, implementado mediante tecnologías estándar de conectividad en Java, JDBC, permite simplificar la arquitectura del sistema para este tipo de cliente y reducir la latencia asociada a la comunicación a través de servicios intermedios.

No obstante, esta arquitectura implica un mayor grado de acoplamiento entre la aplicación de escritorio y el modelo de datos, lo que puede limitar la flexibilidad ante modificaciones en el esquema de la base de datos. Por este motivo, el acceso directo se ha restringido exclusivamente a la aplicación de escritorio y a un entorno controlado, evitando su uso en clientes móviles o en escenarios con mayores requisitos de interoperabilidad.

En conjunto, la implementación de ambos mecanismos de acceso a los datos, conexión directa a la base de datos para la aplicación de escritorio y API REST para la aplicación móvil, permite adaptar la arquitectura del sistema a las necesidades específicas de cada tipo de cliente. Esta estrategia, combinada con la gestión de roles y permisos, contribuye a reforzar la integridad, la eficiencia y la coherencia global del sistema.

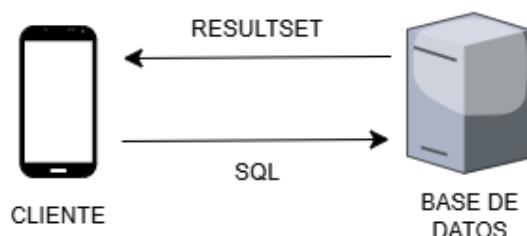


Figura 4: Comunicación directa entre la aplicación de escritorio y la base de datos

## 4 ANÁLISIS DEL SISTEMA

### 4.1 Descripción general del sistema

El sistema desarrollado tiene como objetivo servir de herramienta de apoyo para la gestión de un almacén de consolidación en un entorno industrial. La aplicación permite centralizar la información relacionada con productos, palets, ubicaciones, movimientos y pedidos, facilitando la operativa diaria del almacén y el seguimiento de la mercancía a lo largo de los distintos procesos internos.

La solución se concibe como un sistema distribuido que da soporte a distintos perfiles de usuario mediante interfaces diferenciadas. Por un lado, una aplicación de escritorio orientada a tareas de gestión, planificación y supervisión; por otro, una aplicación móvil destinada a la operativa en planta, permitiendo el acceso a la información y el registro de acciones directamente en el entorno de trabajo.

El sistema está específicamente diseñado para almacenes de consolidación, donde se agrupan mercancías procedentes de distintos proveedores con el fin de preparar envíos conjuntos a clientes. En consecuencia, el alcance funcional se centra en procesos como la gestión de inventario, la consolidación de pedidos y la trazabilidad de los movimientos internos, sin abordar otros tipos de almacén ni el control directo de sistemas automatizados.

### 4.2 Identificación de usuarios

Funcionalidad	SysAdmin	Gestor Almacén	Operario	Administración
Acceso a la gestión general del almacén	Sí	Sí	Sí	No
Consulta y gestión del inventario	Sí	Sí	Sí	No
Gestión de pedidos	Sí	Sí	No	Sí
Paletización de mercancía	Sí	Sí	Sí	No
Gestión de envíos	Sí	Sí	Sí	No
Consulta del calendario de operaciones	Sí	Sí	Sí	Sí
Registro de movimientos de mercancía	Sí	Sí	Sí	No
Actualización de información de palets	Sí	Sí	No	No
Gestión de órdenes de compra	Sí	Sí	No	Sí
Exportación de datos del sistema	Sí	No	No	Sí
Creación de pedidos	Sí	Sí	No	Sí
Edición de pedidos	Sí	Sí	No	Sí
Eliminación de pedidos	Sí	Sí	No	Sí
Creación de productos	Sí	Sí	No	No
Creación de tipos de producto	Sí	Sí	No	No
Creación de usuarios	Sí	No	No	Sí
Edición de usuarios	Sí	No	No	Sí
Eliminación de usuarios	Sí	No	No	Sí

Tabla 1: Permisos de acceso a funcionalidades del sistema según rol de usuario.

El sistema contempla distintos tipos de usuarios o roles, definidos en función de su rol dentro del almacén. La aplicación de los roles se aplica exclusivamente a la aplicación de escritorio, donde se concentran las funcionalidades de gestión, administración y supervisión del sistema. En la aplicación móvil Android, orientada a la operativa en planta, todos los usuarios actúan bajo un perfil funcional equivalente al de operario, independientemente de su rol dentro de la organización.

En la aplicación de escritorio se han definido cuatro roles principales: Administrador del sistema (SysAdmin), Gestor de almacén, Operario y Administración. Cada uno de estos roles dispone

de un conjunto específico de permisos que determinan el acceso a las distintas funcionalidades del sistema. La Tabla 1 recoge de forma resumida los permisos asignados a cada rol en relación con las funcionalidades disponibles. Por ejemplo: Un SysAdmin puede crear, editar y eliminar usuarios, mientras que un Operario no tiene acceso a estas funciones.

El rol de SysAdmin dispone de acceso completo a todas las funcionalidades del sistema, incluyendo la gestión de usuarios, la configuración general, la administración de productos, pedidos y la exportación de datos. Este perfil está orientado a tareas de administración avanzada y mantenimiento del sistema.

El Gestor de almacén es responsable de la planificación y supervisión de la operativa diaria. Este rol puede gestionar inventario, pedidos, palets, envíos y órdenes de compra, así como realizar tareas relacionadas con la organización del almacén, pero no dispone de permisos para la gestión de usuarios ni para la exportación de datos del sistema.

El rol de Administración está orientado a tareas de carácter administrativo y documental. Este perfil puede gestionar pedidos, órdenes de compra, usuarios y exportar información del sistema, pero no interviene directamente en la operativa física del almacén ni en la gestión de movimientos de mercancía.

Por último, el Operario de almacén se encarga de ejecutar las tareas operativas relacionadas con la manipulación y movimiento de mercancías. En la aplicación de escritorio, su acceso se limita a funcionalidades operativas como la consulta de información, la paletización, el registro de movimientos y la gestión de envíos, sin permisos para tareas de administración o gestión avanzada. En la aplicación móvil Android, este rol constituye el único perfil de uso, permitiendo registrar movimientos, consultar inventario y apoyar los procesos de consolidación directamente en planta.

Esta estructura de roles y permisos permite limitar las funcionalidades y las interfaces del sistema a las responsabilidades de cada tipo de usuario, mejorando la eficiencia operativa, reforzando el control de acceso y reduciendo la probabilidad de errores derivados del uso indebido de funcionalidades no autorizadas.

## 4.3 Requisitos del sistema

### 4.3.1 Requisitos funcionales

El sistema debe cumplir los siguientes requisitos funcionales:

- Permitir la gestión de productos y sus características asociadas.
- Gestionar palets y su ubicación dentro del almacén.
- Registrar movimientos internos de mercancía y cambios de ubicación.
- Facilitar la preparación y consolidación de pedidos.
- Permitir el acceso al sistema a distintos perfiles de usuario.
- Proporcionar interfaces diferenciadas para tareas de gestión y operativa en planta.
- Consultar el estado del inventario y la trazabilidad de los productos.

### 4.3.2 Requisitos no funcionales

Además de los requisitos funcionales, el sistema debe cumplir una serie de requisitos no funcionales:

- El sistema debe ser fácil de usar por personal con distintos niveles de experiencia.
- La información debe mantenerse consistente y actualizada en todo momento.
- El acceso a los datos debe estar controlado según el perfil de usuario.

- El sistema debe ser mantenible y permitir futuras ampliaciones funcionales.
- La solución debe ser compatible con entornos industriales habituales.

#### 4.4 Restricciones técnicas

El desarrollo del sistema ha estado condicionado por una serie de restricciones técnicas y de alcance propias de un Trabajo Fin de Grado.

En primer lugar, la solución no contempla el control directo de maquinaria ni de sistemas de automatización industrial, centrándose exclusivamente en la gestión de la información asociada al almacén. Asimismo, el sistema ha sido diseñado específicamente para un entorno de almacén de consolidación, por lo que no resulta aplicable de forma directa a otros tipos de almacén con procesos logísticos diferentes.

Por último, se han considerado las limitaciones temporales y de recursos inherentes a un Trabajo Fin de Grado, desarrollado por un único autor y dentro de un periodo de tiempo acotado. Como consecuencia, se ha priorizado la implementación de las funcionalidades esenciales para la gestión del almacén de consolidación, quedando fuera del alcance del proyecto aspectos como la integración con sistemas externos, el control directo de equipos automatizados o el desarrollo de funcionalidades avanzadas de optimización.

## 5 DISEÑO DEL SISTEMA

En este capítulo se describe el diseño del sistema desarrollado, detallando la estructura general del sistema, la estructura de sus distintos componentes y clases, así como las decisiones adoptadas para implementar cada una de las funcionalidades. El objetivo de este capítulo es definir cómo se materializan, desde un punto de vista técnico, los requisitos identificados en la sección 4.3, sirviendo de base para la posterior fase de implementación.

### 5.1 Arquitectura general del sistema

El sistema ha sido diseñado siguiendo una arquitectura distribuida de tipo cliente-servidor, en la que los estaciones de trabajo fijas y teléfonos móviles (clientes) interactúan con el servidor central.

La solución está compuesta por tres elementos principales: una aplicación de escritorio, una aplicación móvil Android y un servidor central MySQL que contiene la base de datos. Ambos clientes operan dentro de una red interna privada del almacén, lo que permite un entorno controlado desde el punto de vista de seguridad y acceso.

La aplicación móvil Android se comunica con el servidor mediante una API de tipo REST, mientras que la aplicación de escritorio establece una conexión directa con la base de datos. Esta separación responde a criterios de simplicidad, rendimiento y adecuación al entorno operativo de cada dispositivo que empleamos como cliente.

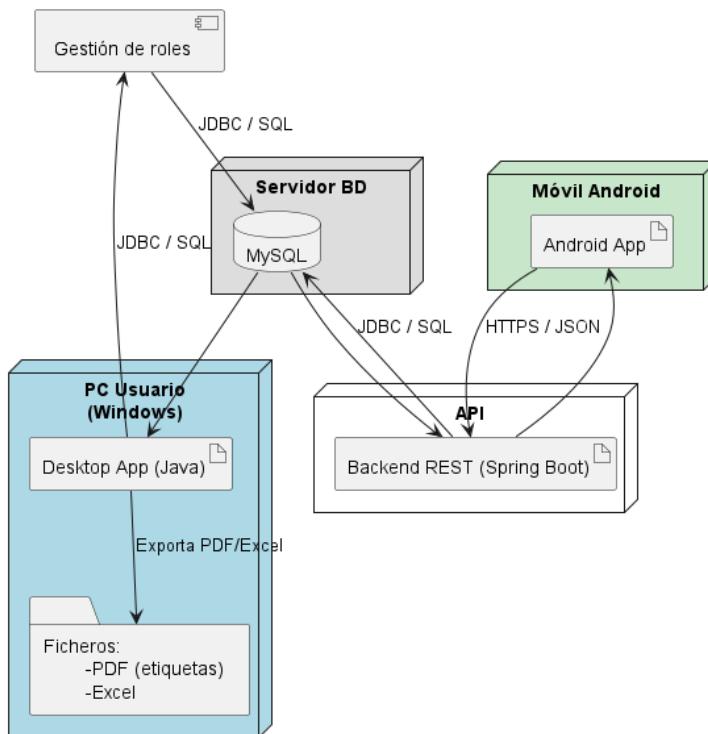


Figura 5: Arquitectura general del sistema cliente-servidor

### 5.2 Diseño de la aplicación Desktop

La aplicación de escritorio ha sido diseñada como la herramienta principal de gestión y administración del sistema. Su diseño está orientado a usuarios con privilegios elevados, como administradores del sistema, gestores de almacén y personal administrativo.

# DISEÑO DE UNA APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE UN ALMACÉN SIN AUTOMATIZACIÓN CON DISTINTAS INTERFAZES PARA LOS OPERARIOS

JOSÉ TOMÁS TORRE PEDROARENA

Desde un punto de vista estructural, la aplicación se organiza en una ventana principal desde la cual se accede a las distintas funcionalidades mediante botones y menús. Algunas acciones se realizan dentro de la propia ventana principal, mientras que otras abren ventanas secundarias especializadas para tareas concretas, como la gestión de productos, pedidos o usuarios.

La lógica de la aplicación se estructura siguiendo una separación entre la capa de presentación, encargada de la interfaz gráfica, y la capa de acceso a datos, responsable de la interacción con la base de datos.

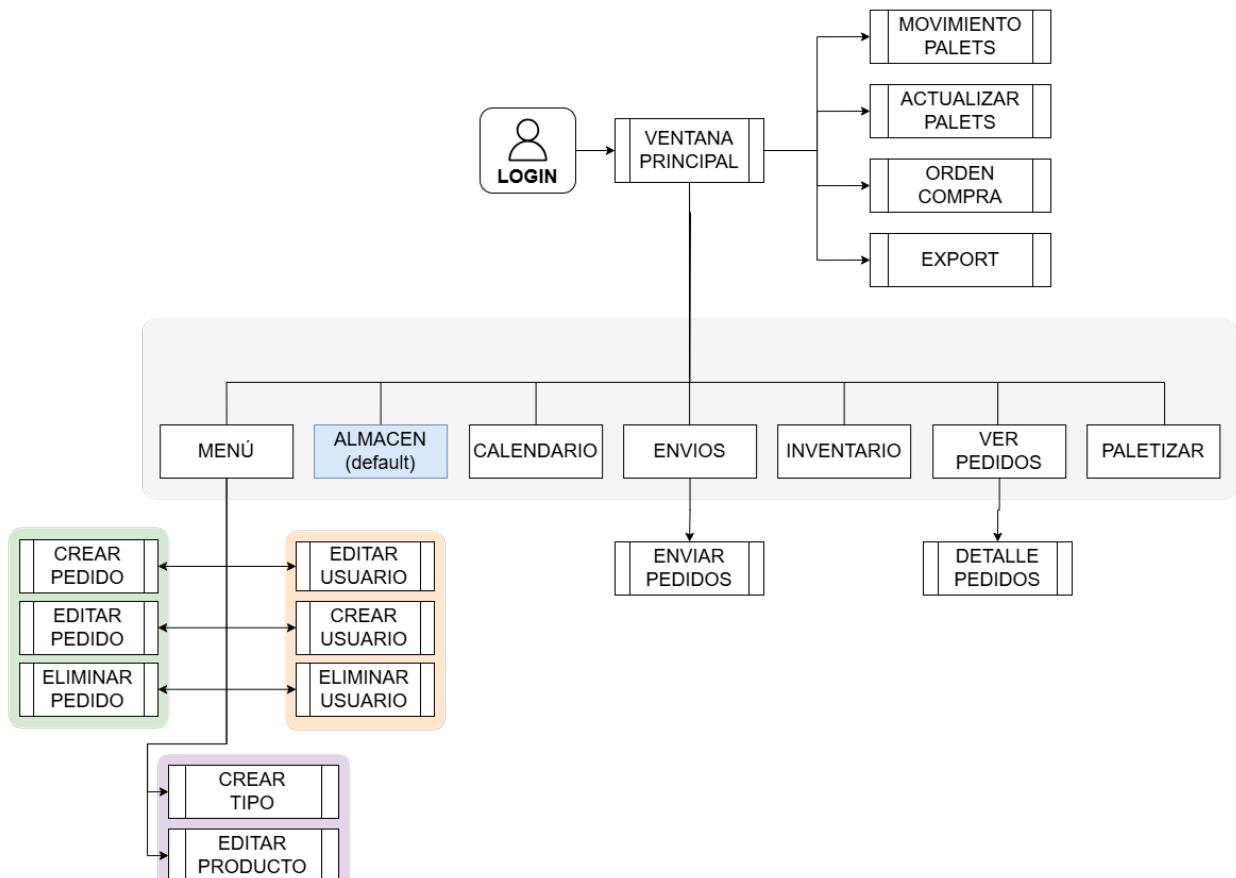


Figura 6: Diagrama de navegación entre ventanas de la aplicación Desktop

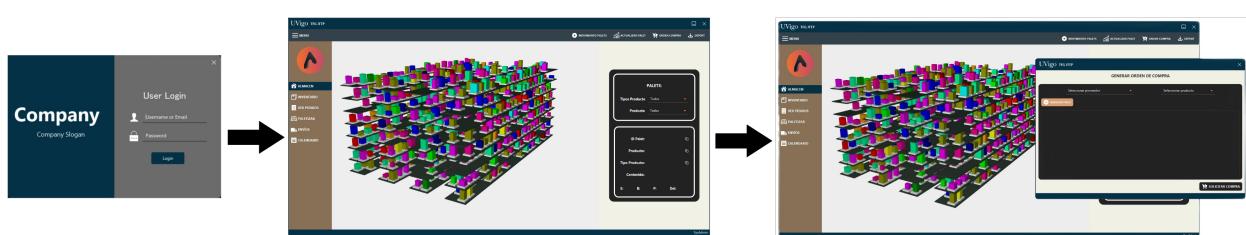


Figura 7: Ejemplo de navegación entre ventanas de la aplicación Desktop (desde *Login* hasta *Orden Compras*)

## 5.3 Diseño de la aplicación Android

La aplicación móvil Android ha sido diseñada como una herramienta de apoyo a la operativa en planta, orientada principalmente a los operarios del almacén. Su diseño prioriza la simplicidad de uso y el acceso rápido a la información relevante durante la ejecución de tareas.

La aplicación actúa como un cliente ligero que delega la lógica de negocio en el servidor,

comunicándose con este mediante una API REST. De este modo, la aplicación móvil se limita a gestionar la interfaz de usuario y el envío y recepción de datos.

Las distintas pantallas de la aplicación se organizan en torno a las tareas habituales del operario, como la consulta de inventario, el registro de movimientos o la gestión de palets.

## 5.4 Diseño de la base de datos

La base de datos ha sido diseñada para almacenar de forma estructurada la información necesaria para la gestión del almacén de consolidación. El modelo de datos contempla entidades como productos, palets, ubicaciones, pedidos, movimientos y usuarios.

El diseño sigue un enfoque relacional, garantizando la integridad referencial y evitando redundancias innecesarias. Se han definido claves primarias y foráneas para representar las relaciones entre las distintas entidades, así como restricciones que aseguran la coherencia de los datos almacenados.

## 5.5 Diseño de la API

La API REST ha sido diseñada para permitir la comunicación entre la aplicación móvil Android y el backend del sistema. Esta API expone un conjunto de recursos que representan las entidades principales del sistema, como productos, palets, pedidos y movimientos.

Cada recurso se gestiona mediante operaciones HTTP estándar, permitiendo realizar consultas, inserciones y actualizaciones de forma controlada. El diseño de la API busca mantener una estructura clara y coherente, facilitando su mantenimiento y posible ampliación futura.

## 5.6 Diagramas UML

Para representar de forma gráfica la estructura y el comportamiento del sistema, se han empleado distintos diagramas UML. Estos diagramas permiten visualizar las relaciones entre clases, la interacción entre componentes y el despliegue físico del sistema.

### 5.6.1 Diagrama de clases

El diagrama de clases representa las principales entidades del sistema, sus atributos y las relaciones existentes entre ellas. Este diagrama sirve como referencia para el diseño del modelo de datos y para la implementación de la lógica del sistema.

### 5.6.2 Diagrama de secuencia

El diagrama de secuencia describe la interacción entre los distintos componentes del sistema durante la ejecución de casos de uso representativos, como el registro de un movimiento o la creación de un pedido.

### 5.6.3 Diagrama de despliegue

El diagrama de despliegue muestra la distribución física de los componentes del sistema, incluyendo clientes, servidor y base de datos, así como los canales de comunicación entre ellos.

## 5.7 Diseño de la interfaz de usuario

El diseño de la interfaz de usuario se ha planteado teniendo en cuenta los distintos perfiles de usuario y los contextos de uso del sistema. En la aplicación de escritorio se prioriza la claridad en la presentación de la información y el acceso estructurado a las funcionalidades de gestión.

**DISEÑO DE UNA APLICACIÓN INFORMÁTICA PARA LA GESTIÓN DE UN ALMACÉN SIN AUTOMATIZACIÓN CON DISTINTAS INTERFACES PARA LOS OPERARIOS**

**JOSÉ TOMÁS TORRE PEDROARENA**

---

En la aplicación móvil, el diseño se orienta a la rapidez de uso y a la reducción de acciones necesarias para completar una tarea. Para definir la estructura visual de ambas aplicaciones se han utilizado bocetos y esquemas previos, que han servido como guía durante la implementación.

## 6 IMPLEMENTACIÓN

### 6.1 Herramientas y tecnologías utilizadas

El sistema desarrollado se ha implementado íntegramente utilizando el lenguaje de programación Java. Este lenguaje se ha empleado tanto en el desarrollo de la aplicación de escritorio como en la aplicación móvil Android y en el backend del sistema, implementado mediante el framework Spring Boot.

La utilización de un lenguaje común en todos los componentes de la solución facilita la integración entre las distintas aplicaciones, simplifica el mantenimiento del sistema y permite reutilizar conocimientos y conceptos a lo largo de todo el desarrollo. Además, Java es un lenguaje ampliamente utilizado en entornos industriales y empresariales, caracterizado por su portabilidad, robustez y amplio ecosistema de herramientas y librerías **oracle\_java**.

- **Control de versiones:** Git.
- **Backend:** Java junto con el framework Spring Boot.
- **Base de datos:** MySQL.
- **Gestión y modelado de datos:** MySQL Workbench y Draw.io.
- **Desarrollo de la aplicación de escritorio:** IntelliJ IDEA Ultimate.
- **Desarrollo de la aplicación móvil:** Android Studio.
- **Pruebas de la API:** Postman.
- **Entorno de despliegue:** VMware Workstation con un servidor Ubuntu.
- **Gestión del proyecto:** Git.

### 6.2 Implementación de la aplicación Desktop

### 6.3 Implementación de la aplicación Android

### 6.4 Implementación del backend / servidor

### 6.5 Implementación de la base de datos

### 6.6 Seguridad y gestión de errores

### 6.7 Control de versiones y gestión del proyecto

---

## 7 RESULTADOS Y DISCUSIÓN

### 7.1 Resultados del desarrollo

### 7.2 Evaluación del cumplimiento de objetivos

### 7.3 Ventajas de la solución propuesta

### 7.4 Limitaciones detectadas

---

## 8 CONCLUSIONES Y TRABAJOS FUTUROS

### 8.1 Conclusiones

### 8.2 Líneas de mejora y trabajos futuros

## REFERENCIAS

- [1] Universidad de Vigo. "Arquitectura Cliente/Servidor." Material docente de la asignatura Sistemas Cliente/Servidor. Último acceso: enero 2026. dirección: <https://ccia.esei.uvigo.es/docencia/SCS/1011/transparencias/Tema1.pdf>
- [2] Comisión Nacional de los Mercados y la Competencia. "Panel de hogares: usos de Internet." Gráfica "Sistema operativo del smartphone". Último acceso: enero 2026. dirección: <https://www.cnmc.es/prensa/panel-hogares-usos-internet-20231103>
- [3] Google. "Android Developers Documentation," visitado ene. de 2026. dirección: <https://developer.android.com/guide/components/fundamentals>
- [4] Asociación Española de Banca. "APIs." Documento institucional. Último acceso: enero 2026. dirección: <https://s1.aebanca.es/wp-content/uploads/2018/05/apis.pdf>
- [5] Iniciativa Aporta. "Buenas prácticas en el diseño de APIs y linked data." Gobierno de España. dirección: [https://datos.gob.es/elearning/Unidades\\_Didacticas/Unidad\\_8/contenidos/descargas/unidad\\_imprimible.pdf](https://datos.gob.es/elearning/Unidades_Didacticas/Unidad_8/contenidos/descargas/unidad_imprimible.pdf)
- [6] Amazon Web Services. "¿Qué es un marco en programación e ingeniería?" Visitado ene. de 2026. dirección: <https://aws.amazon.com/what-is/framework/>
- [7] Oracle. "What Are Backends?" Visitado ene. de 2026. dirección: <https://docs.oracle.com/en/cloud/paas/app-builder-cloud/visual-builder-developer/what-are-backends.html>
- [8] VMware. "Spring Boot Reference Documentation," visitado ene. de 2026. dirección: <https://docs.spring.io/spring-boot/docs/current/reference/html/>