
Especificación de requisitos de software

Para

AUTOMOTIVE COMPANY

Versión 1.0 aprobado

**Preparado por yosser caraballo, jean Carlos mórelo, Sebastián
Paternina**

<organización>

17/03/2025

Tabla de contenido

	<i>Page</i>
1. Introducción	1
2. Descripción general	2
2.1 Características del producto	2
2.2 Clases de usuarios y características	2
2.3 Limitaciones de diseño y aplicación	2
2.4 Documentación del Usuario	3
2.5 Supuestos y dependencias	3
3. Características del sistema	4
3.1 Características de Sistema	4
3.1.1 Descripción y prioridad	4
3.1.2 Secuencias de estímulo/respuesta	4
3.1.3 Requisitos funcionales	4
4. Requisitos de la interfaz	6
4.1 Interfaces de Usuario	6
4.2 Interfaces de hardware	6
4.3 Interfaces de software	6
4.4 Interfaces de Comunicación	6
5. Otros requisitos No funcionales	7
5.1 Requerimientos de desempeño	7
5.2 Requisitos de seguridad	7
5.3 Atributos de calidad del software	7
6. Otros Requerimientos	

8 7. Referencias
..... **9**

Historial de revisiones

Nombre	Fecha	Motivo de los cambios	Versión
yosser	17/03/2025	Creación del MVP	0.1
Jean Carlos morelo	17/03/2025	-Cambio en la introducción -Creación de la descripción general -Creacion de las características del producto -Definicion de los requerimientos funcionales	0.11
Jean carlos morelo	23/03/2025	-cambios en los requerimientos para la especificación del mvp y características	0.12
sebastian paternina	23/03/2025	- revisión y mejora de los requerimientos	0.13
Yosser Caraaballo	26/04/2025	- Se integro servicios CRUD	0.14
Jean Carlos morelo	7/05/2025	Cambios en toda la parte de requisito de la interfaz 4.1, 4.2, 4.3 y 4.4 respectivamente, Y cambios en la parte de características del producto, 2.2, 2.3 y 2.5 exactamente	0.15
Jean Carlos morelo	05/06/2025	Cambios en descripción general, Apendice C y características del sistema	1.0

1. Introducción

<Software para gestión de una empresa automotriz centrado en la creación, almacenamiento, y distribución de carros los beneficios de este software es una mejor organización al permitir una gestión en tiempo real de los pedidos y la disponibilidad del stock permitiendo una mayor eficiencia a la hora de trabajar, siendo el objetivo principal del software una mejora significativa en la organización y la eficiencia en el manejo de pedidos y stock por medio de un software con APIs bien definidas

2. Descripción general

Perspectiva del producto

Un software que ayuda en la organización del inventario de una compañía automotriz por medio de opciones de pedidos, inventario y un buscador para facilitar la búsqueda en específico de cierto componente/pedido/auto permitiendo una fácil organización del inventario y el control sobre el mismo en la compañía.

<este producto fue creado en el contexto de que una empresa automotriz se manejaba por medio de diferentes sistemas para secciones como el diseño del vehículo, la fabricación, el inventario etc... pero en el proceso se limitaba la información porque los datos se encuentran aislados en silos esto siendo un producto de la fragmentación de los sistemas provocando que no se pueda visibilizar el stock y generando reservas excesivas, pedidos duplicados y capital inflado. este software se creó como respuesta a esta problemática un programa que comparta datos multi funcionalmente para evitar estas problemáticas por medio de microservicios como la gestión de inventario y la fabricación, optimizando el proceso y evitando las problemáticas del sistema anterior>

2.1 Características del producto

- <-Micro servicios que involucran la gestión de inventario y la fabricación.*
 - API de datos unificadas gestionadas por una puerta enlace.*
 - Gestión del stock en tiempo real.*
 - Gestión y consulta de pedidos realizados y disponibilidad del stock(disponible, no disponible o estado del pedido etc...).*
 - Sistema de búsqueda basado en una ID asignada.*
 - Sistema que permite realizar pedidos y asignar la ID correspondiente al pedido*
- >*

2.2 Clases de usuarios y características

- Uso medio. Conocimientos técnicos altos. Acceso solo lectura.

Personal de Soporte:

- Mantenimiento y resolución de fallos.
- Uso puntual. Perfil técnico. Acceso total.

<Identifique las distintas clases de usuarios que prevé que utilizarán este producto. Las clases de usuarios pueden diferenciarse en función de la frecuencia de uso, el subconjunto de funciones del producto utilizadas, los conocimientos técnicos, los niveles de seguridad o privilegios, el nivel educativo o la experiencia. Describa las características pertinentes de cada clase de usuario. Algunos requisitos pueden pertenecer sólo a determinadas clases de usuarios. Distinga las clases de usuarios favorecidas de las que son menos importantes de satisfacer.>

2.3 Limitaciones de diseño y aplicación

Requisitos técnicos:

- Framework: Flutter (frontend)
- Backend: Python
- Base de datos: PostgreSQL
- Plataforma de despliegue: Fly.io

Limitaciones:

- Arquitectura distribuida basada en microservicios.
- Estándares de seguridad y comunicación cifrada (HTTPS).
- Integración con sistemas heredados.
- Uso obligatorio del API Gateway.

<Describa los elementos o problemas que limitarán las opciones disponibles para los desarrolladores. Estos pueden ser: políticas corporativas o normativas; limitaciones de hardware (requisitos de tiempo, requisitos de memoria); interfaces con otras aplicaciones; tecnologías, herramientas y bases de datos específicas que se vayan a utilizar; operaciones paralelas; requisitos de lenguaje; protocolos de comunicación; consideraciones de seguridad; convenciones de diseño o normas de programación (por ejemplo, si la organización del cliente será responsable del mantenimiento del software entregado).>

2.4 Documentación del Usuario

<Enumere los componentes de la documentación de usuario (como manuales de usuario, ayuda en línea y tutoriales) que se entregarán junto con el software. Identifique los formatos o normas de entrega de documentación de usuario conocidos.>

2.5 Supuestos y dependencias

Fly.io estará disponible durante la operación.

- PostgreSQL se mantendrá como base de datos.
- Habrá conectividad entre microservicios y API Gateway.
- Se podrán integrar datos legados con procesos ETL.
- Los usuarios contarán con navegadores modernos.

< Enumere los factores supuestos (en lugar de los hechos conocidos) que podrían afectar a los requisitos establecidos en la SRS. Pueden ser componentes comerciales o de terceros que se vayan a utilizar, problemas relacionados con el desarrollo o el entorno operativo, o limitaciones. El proyecto podría verse afectado si estos supuestos son incorrectos, no se comparten o cambian. Identifique también cualquier dependencia que tenga el proyecto de factores externos, como componentes de software que pretenda reutilizar de otro proyecto, a menos que ya estén documentados en otra parte (por ejemplo, en el documento de visión y alcance o en el plan del proyecto).>

3. Características del sistema

<Esta plantilla ilustra la organización de los requisitos funcionales del producto por características del sistema, los principales servicios proporcionados por el producto. Es posible que prefiera organizar esta sección por caso de uso, modo de funcionamiento, clase de usuario, clase de objeto, jerarquía funcional o combinaciones de estos, lo que tenga más sentido para su producto.>

3.1 Características de Sistema

Inventario: una forma simple de organizar el inventario de la compañía con este presentando nombre, id, estado, etc...

Agregar,; es la función simple de agregación de un componente o vehículo en el inventario

<No diga realmente "Característica del sistema 1". Indique el nombre de la característica en pocas palabras.>

3.1.1 Descripción y prioridad

<Presente una breve descripción de la característica e indique si es de alta, media o baja prioridad. También puede incluir calificaciones de componentes prioritarios específicos, como el beneficio, la penalización, el coste y el riesgo (cada uno de ellos calificado en una escala relativa que va de un 1 a un 9).>

3.1.2 Secuencias de estímulo/respuesta

[Usuario] Abre la app.

[Sistema] Muestra la interfaz de inicio.

[Usuario] Presiona el botón inventario.

[Sistema] Muestra los datos del inventario junto a su interfaz.

[Usuario] Presiona el botón agregar.

[Sistema] interfaz para la creación de un componente en pedido.

[Usuario] Agrega los datos.

[Sistema] Agrega el pedido/componente/vehículo a la base de datos para que se muestre en el inventario.

<Lista las secuencias de acciones del usuario y las respuestas del sistema que estimulan el comportamiento definido para esta característica. Estos corresponderán a los elementos de diálogo asociados a los casos de uso.>

3.1.3 Requisitos funcionales

MVP:

REQ-1: Registro del stock.

- Debe actualizar en tiempo real el stock y mostrarlo por mediante de una GUI mostrando exactamente cuánto stock hay de cada artículo esté stock siendo visible para los administradores así teniendo la capacidad de administrar el trabajo de mejor manera.

REQ-2: Sistema de pedidos

- Debe recibir, notificar, al usuario y a la empresa de la compra del vehículo
- Brindar información del estado y ubicación del vehículo.

REQ-3: Sistema de consulta de stock.

- Los usuarios deberían poder consultar sobre la disponibilidad de un producto en específico y saber cuánto stock tiene.

REQ-4: Sistema de consulta de pedidos.

- Los usuarios deben tener la opción de poder consultar sus pedidos por medio de una GUI en la cual muestre sus pedidos pendientes mostrando fechas, estado del pedido y el lugar al que se va a entregar.
- Buscar por el ID del pedido.

REQ-5: Modificación y cancelación de pedidos

- Modificar el pedido de una pieza, pero primero debe pasar por una aprobación por el sistema de administradores y así cambiar la fecha por el cambio abrupto no debería permitir hacer una modificación o eliminación 7 días antes de que se termine y entregue el pedido.

REQ-6: Registro y actualización sobre entrega de proveedores.

- Permitir a los administradores tener un registro completo el cual se actualizo de la entrega y la reposición de stock por parte de los proveedores.

REQ-7: Registro usuario

- Permitir al usuario registrarse con un usuario y contraseña y crear una cuenta en la cual se mantiene el registro de pedidos que ha hecho esto permite la identificación de un usuario y también la organización más eficiente de pedidos y control de estos.

REQ-8: Lago de datos.

- Permite unificar los datos de los diferentes sistemas teniendo datos multifuncionales arreglando el problema de fragmentación y permitiendo la visión integral de la cadena de valor automotriz.

4. Requisitos de la interfaz

4.1 Interfaces de Usuario

El software será una aplicación ****web****, accesible a través de navegadores modernos. Estará diseñada con un enfoque minimalista, utilizando una ****paleta de colores simple con predominancia del negro**** y una interfaz clara para facilitar el uso en entornos relacionados con la industria automotriz.

Componentes principales de la interfaz:

- Inventario:

Muestra una tabla con todos los productos disponibles, su ID, nombre, cantidad en stock y distribuidor. Al seleccionar un producto se accede a una vista detallada con datos adicionales como stock específico, proveedor, fecha de ingreso, y observaciones internas.

- Búsqueda:

Funcionalidad clave para localizar productos o pedidos por su ID única. Al ingresar una ID válida se mostrará un panel con toda la información relacionada. Esto agiliza el acceso directo a datos sin necesidad de navegación manual.

- Pedidos:

Sección para visualizar pedidos realizados. Se incluyen columnas como ID del pedido, fecha, estado (pendiente, en proceso, entregado). Permite hacer seguimiento detallado de cada pedido.

Características generales de la interfaz:

- * Navegación principal mediante barra superior con secciones fijas: Inventario, Búsqueda, Pedidos.
 - * Botones estándar: `Guardar`, `Cancelar`, `Editar`, `Eliminar`, `Atrás`.
 - * Mensajes de error claros, por ejemplo: “Producto no encontrado”, “Cantidad fuera de stock”.
 - * Diseño adaptable a pantallas de escritorio.
 - * Accesible mediante navegador (Google Chrome, Mozilla Firefox, Microsoft Edge).
 - * No incluye integración con dispositivos como escáneres o lectores externos.
- El diseño visual completo será desarrollado más adelante y documentado en la *Especificación de Interfaz de Usuario* correspondiente.

<Describe las características lógicas de cada interfaz entre el producto de software y los usuarios. Esto puede incluir imágenes de pantalla de muestra, cualquier estándar de interfaz gráfica de usuario

o guías de estilo de la familia de productos que deban seguirse, restricciones de diseño de pantalla, botones y funciones estándar (por ejemplo, ayuda) que aparecerán en cada pantalla, atajos de teclado, estándares de visualización de mensajes de error, etc. Defina los componentes del software para los que se necesita una interfaz de usuario. Los detalles del diseño de la interfaz de usuario deben documentarse en una especificación de interfaz de usuario aparte.>

4.2 Interfaces de hardware

El sistema no requiere integración con hardware especializado. Está pensado para usarse en oficinas administrativas de una compañía automotriz, por lo que solo necesita:

- PCs o laptops de uso común con teclado, ratón y acceso a internet.
- Navegador web actualizado.
- Compatible con Windows, macOS y distribuciones comunes de Linux.

<Describa las características lógicas y físicas de cada interfaz entre el producto de software y los componentes de hardware del sistema. Esto puede incluir los tipos de dispositivos admitidos, la naturaleza de las interacciones de datos y control entre el software y el hardware, y los protocolos de comunicación que se utilizarán.>

4.3 Interfaces de software

El sistema interactuará con los siguientes componentes de software:

- Base de datos PostgreSQL (versión 13 o superior): utilizada para almacenar toda la información de productos, pedidos, usuarios y registros del sistema.
- Sistema operativo del servidor: Linux o windows.

- Framework de desarrollo web (flutter).
- Frontend: tecnologías web modernas (flutter).

Intercambio de datos:

- Comunicación entre frontend y backend mediante API RESTful.
- Formato de intercambio: JSON.
- Elementos clave: ID del producto, nombre, cantidad, distribuidor, estado del pedido, fecha de creación.

<Describe las conexiones entre este producto y otros componentes de software específicos (nombre y versión), incluyendo bases de datos, sistemas operativos, herramientas, bibliotecas y componentes comerciales integrados. Identifique los elementos de datos o mensajes que entran y salen del sistema y describa el propósito de cada uno. Describa los servicios necesarios y la naturaleza de las comunicaciones. Consulte los documentos que describen los protocolos detallados de la interfaz de programación de aplicaciones. Identifique los datos que se compartirán entre los componentes del software. Si el mecanismo de compartición de datos debe implementarse de una manera específica (por ejemplo, el uso de un área de datos global en un sistema operativo multitarea), especifíquelo como una restricción de implementación.>

4.4 Interfaces de Comunicación

El sistema requiere conexión a red local o internet para funcionar, pero **no incluye funcionalidades de correo electrónico ni formularios electrónicos**.

Requisitos de comunicación:

- Comunicación entre cliente y servidor mediante protocolo HTTPS.

- Interacción con base de datos a través de backend.
- No se requiere FTP, SMTP ni otros protocolos de comunicación externos.
- Seguridad básica garantizada mediante:
 - Cifrado de datos en tránsito (HTTPS con SSL/TLS).
 - Autenticación de usuario con contraseñas encriptadas.
 - Control de acceso por niveles o roles de usuario.

<Describa los requisitos asociados a cualquier función de comunicación que requiera este producto, incluyendo el correo electrónico, el navegador web, los protocolos de comunicación del servidor de red, los formularios electrónicos, etc. Defina cualquier formato de mensaje pertinente. Identifique los estándares de comunicación que se utilizarán, como FTP o HTTP. Especifique las cuestiones de seguridad o encriptación de las comunicaciones, las tasas de transferencia de datos y los mecanismos de sincronización.>

5. Otros requisitos No funcionales

5.1 Requerimientos de desempeño

<Si existen requisitos de rendimiento para el producto en diversas circunstancias, indíquelos aquí y explique su razón de ser, para ayudar a los desarrolladores a entender la intención y tomar las decisiones de diseño adecuadas. Especifique las relaciones de tiempo para los sistemas de tiempo real. Especifique al máximo estos requisitos. Es posible que tenga que establecer requisitos de

rendimiento para requisitos funcionales o características individuales.> Requisitos de seguridad.>

5.2 Requisitos de seguridad

<Especifique cualquier requisito relativo a cuestiones de seguridad o privacidad en torno al uso del producto o a la protección de los datos utilizados o creados por el producto. Defina cualquier requisito de autenticación de la identidad del usuario. Refiérase a cualquier política o normativa externa que contenga cuestiones de seguridad que afecten al producto. Definir cualquier certificación de seguridad o privacidad que deba ser satisfecha.>

5.3 Atributos de calidad del software

<Especifique cualquier característica de calidad adicional para el producto que sea importante para los clientes o los desarrolladores. Algunas de las características a tener en cuenta son: adaptabilidad, disponibilidad, corrección, flexibilidad, interoperabilidad, mantenibilidad, portabilidad, fiabilidad, reutilización, robustez, con probabilidad y usabilidad. Redáctelos de forma específica, cuantitativa y verificable cuando sea posible. Como mínimo, aclare las preferencias relativas de los distintos atributos, como la facilidad de uso sobre la facilidad de aprendizaje.>

6. Otros Requerimientos

<Defina cualquier otro requisito que no esté contemplado en el SRS. Esto podría incluir requisitos de la base de datos, requisitos de internacionalización, requisitos legales, objetivos de reutilización para el proyecto, etc. Añada cualquier sección nueva que sea pertinente para el proyecto.>

7. Servicio CRUD

1. Microservicio de Inventario

Operación Método Descripción

Create POST Añade un artículo

Read all GET Lista todos los artículos

Read one GET Detalle de un artículo

Update PUT Modifica cantidad o atributos

Delete DELETE Elimina un artículo

2. Microservicio de Búsqueda

Operación Método Descripción

Create index POST Crea un nuevo índice

Read index GET Detalle de la configuración del índice

Update index PUT Ajusta parámetros de indexación

Delete index DELETE Elimina un índice

Search GET Ejecuta consulta de texto completo

3. Microservicio de Pedidos

Operación Método Descripción

Create order POST Crea un nuevo pedido

Read all GET Lista todos los pedidos

Read one GET Detalle de un pedido

Update PUT Actualiza estado o datos del pedido

Delete DELETE Cancela un pedido

8. Referencias

<Lista de otros documentos o direcciones web a los que hace referencia esta SRS. Pueden ser guías de estilo de interfaz de usuario, contratos, normas, especificaciones de requisitos del sistema, documentos de casos de uso o un documento de visión y alcance. Proporcione suficiente información para que el lector pueda acceder a una copia de cada referencia, incluyendo el título, el autor, el número de versión, la fecha y la fuente o la ubicación.>

Apéndice A: Glosario

<Defina todos los términos necesarios para interpretar correctamente el SRS, incluidos los acrónimos y las abreviaturas. Puede crear un glosario independiente que abarque varios proyectos o toda la organización, y limitarse a incluir términos específicos de un solo proyecto en cada SRS.>

Apéndice B: Modelos de análisis

<Opcionalmente, incluya cualquier modelo de análisis pertinente, como diagramas de flujo de datos, diagramas de clases, diagramas de transición de estados o diagramas entidad-relación.>

Apéndice C: Lista de problemas

- No se pudo completar gran parte de los micro servicios por problemas al hacer el repositorio
- se tenía pensado agregar búsqueda y pedidos, pero al final solo quedara inventario

<Esta es una lista dinámica de los problemas de requisitos abiertos que quedan por resolver, incluyendo TBDs, decisiones pendientes, información que se necesita, conflictos que esperan ser resueltos, y similares.>