Informe de arquitectura de software

[La flota pepsiman]

[28/09/2025]

Integrantes: Miguel Fazio

Benjamin Riquelme

Índice

[**Introducción 1**](#_iizoqjp1h2bz)

[**Diagramas de casos de uso 2**](#_n946dm8faysr)

[1. Caso de uso general – Gestión de mantenimiento de la flota. 2](#_vd48nxvmkzdh)

[2. Caso de uso específico – Agendar hora de mantención. 3](#_fuwant3atzru)

[3. Caso de uso específico – Actualizar estado de reparación de camión. 4](#_vtxg9z2vatcq)

[**Diagramas de secuencia 5**](#_vco4ml18e15w)

[1. Diagrama de secuencia – Agendar hora de mantención. 5](#_z1juvloq4oz)

[2. Diagrama de secuencia – Actualizar estado de reparación de camión. 6](#_aem7ujc7w2ms)

[**Diagrama de actividades 7**](#_ukzvrpep547j)

[1. Diagrama actividades - Agendar hora de mantención 7](#_317tpcyncn6b)

[**Diagrama de despliegue 8**](#_vjon5pvj2a0k)

[**Conclusión 9**](#_86mnj4ce24oq)

# Introducción

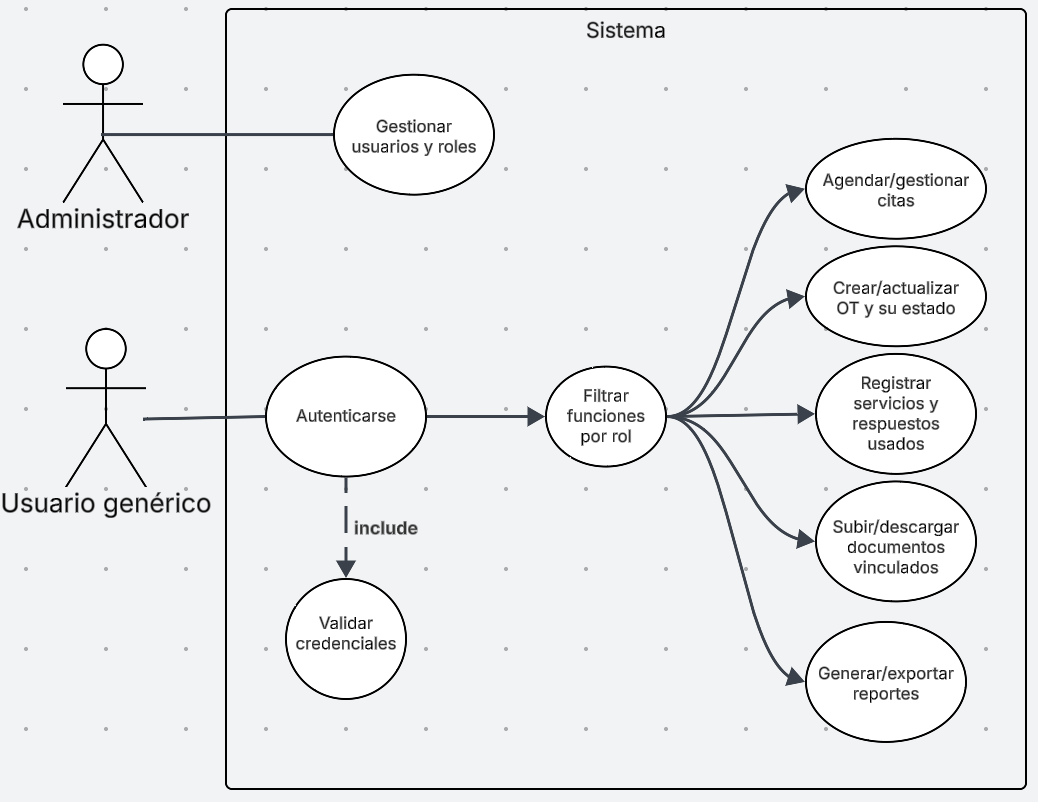
El presente informe recopila los principales diagramas de diseño elaborados durante la fase de diseño de software del sistema *La Flota Pepsiman*.  
Este sistema tiene como propósito apoyar la gestión de mantenimiento de los camiones de la flota Pepsi, permitiendo una administración centralizada de citas, órdenes de trabajo, repuestos, reportes y documentos, bajo un esquema seguro de roles y accesos.

La construcción de los diagramas permite representar de manera clara y estructurada cómo interactúan los diferentes usuarios, procesos y componentes del sistema. Estos modelos servirán como guía de referencia para la siguiente fase de desarrollo e implementación en Laravel con MySQL.

# Diagramas de casos de uso

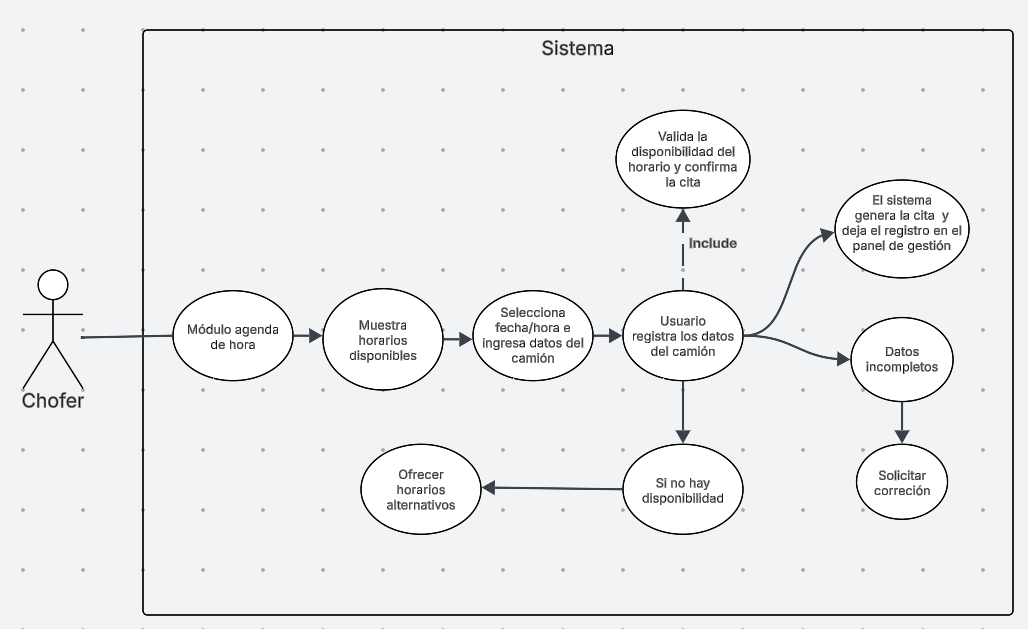
## 1. Caso de uso general – *Gestión de mantenimiento de la flota.*

Este caso de uso general muestra la visión completa del sistema *La Flota Pepsiman*, incluyendo todos los roles de usuario y las funcionalidades principales: agendar y gestionar citas, actualizar el estado de órdenes de trabajo, registrar servicios y repuestos, subir y descargar documentos, generar reportes, así como la gestión de usuarios y roles. Representa la cobertura funcional del sistema y sirve como marco de referencia para los casos de uso específicos.



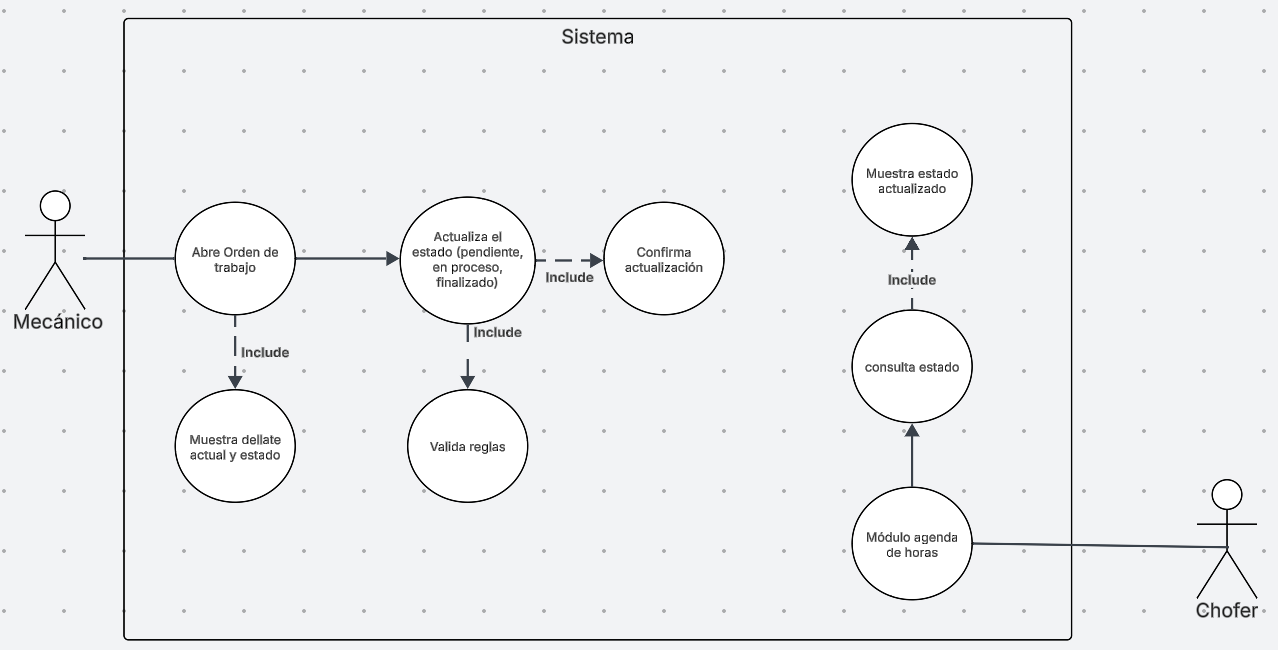
## 2. Caso de uso específico – *Agendar hora de mantención.*

Este caso de uso describe el proceso mediante el cual un chofer o cliente puede acceder al módulo de agenda, visualizar horarios disponibles en el taller, seleccionar fecha y hora, ingresar los datos del camión y confirmar la cita. El sistema valida la disponibilidad y confirma la reserva. Con este caso de uso se representa una de las funcionalidades clave: la organización de tiempos de mantención de los camiones.



## 3. Caso de uso específico – *Actualizar estado de reparación de camión.*

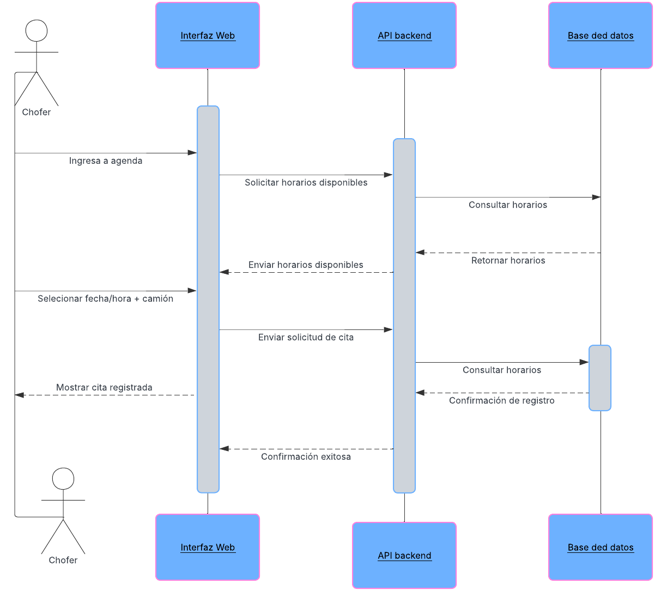
Este caso de uso representa la interacción entre un mecánico y el sistema para actualizar el estado de reparación de un camión (pendiente, en proceso, finalizado). El sistema registra la información en la base de datos y permite que el chofer consulte el estado actualizado desde su módulo de agenda. Con esto se garantiza la trazabilidad del avance en las reparaciones y la comunicación con el cliente.



# Diagramas de secuencia

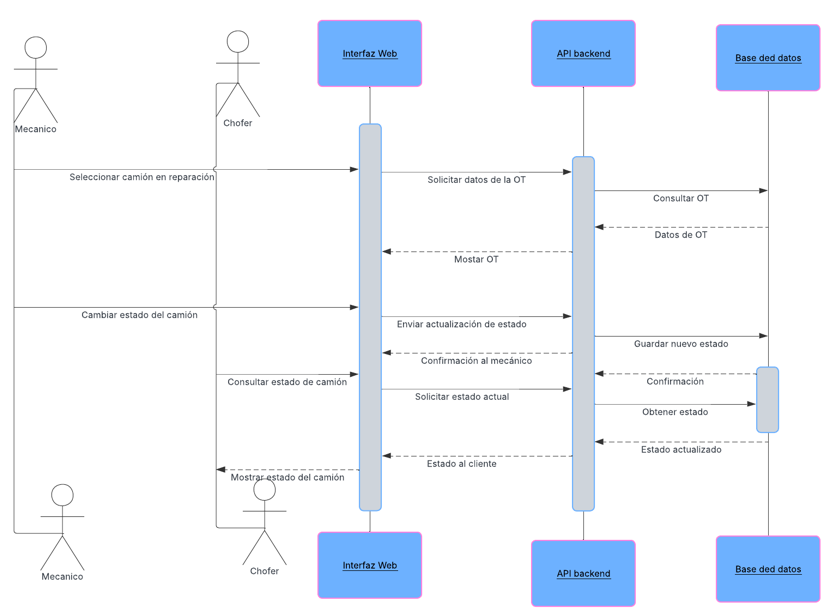
## 1. Diagrama de secuencia – *Agendar hora de mantención.*

Este diagrama muestra la interacción entre el chofer, la interfaz web, el API backend y la base de datos para realizar la acción de agendar una cita.  
 El proceso inicia cuando el chofer accede al módulo de agenda y la interfaz solicita al backend los horarios disponibles. El backend consulta la base de datos y retorna las opciones, que se muestran al chofer. Posteriormente, el usuario selecciona la fecha, hora y camión, enviando la solicitud de cita al sistema. El backend valida y registra la cita en la base de datos, devolviendo una confirmación exitosa que se muestra en la interfaz del chofer.  
 Este diagrama refleja cómo el sistema asegura la coherencia de los horarios y la correcta creación de citas.



## 2. Diagrama de secuencia – *Actualizar estado de reparación de camión.*

Este diagrama describe la interacción entre el mecánico, el chofer, la interfaz web, el API backend y la base de datos.  
 El flujo comienza cuando el mecánico selecciona un camión en reparación, solicitando al backend los datos de la orden de trabajo. Una vez cargada la información, el mecánico actualiza el estado del camión (pendiente, en proceso, finalizado). El backend registra el cambio en la base de datos y confirma al mecánico. Posteriormente, el chofer consulta el estado de su camión; la interfaz solicita esta información al backend, que la obtiene de la base de datos y la muestra actualizada.  
 Este diagrama evidencia cómo el sistema garantiza la trazabilidad del estado del camión y la transparencia hacia el cliente.



# Diagrama de actividades

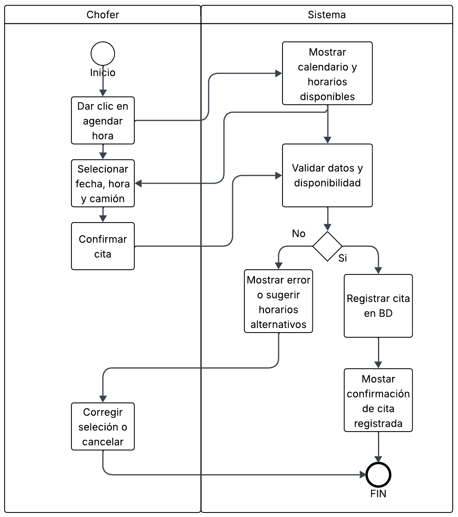
## 1. Diagrama actividades - Agendar hora de mantención

El diagrama de actividades representa el flujo de trabajo que realiza un chofer al momento de agendar una hora de mantención en el sistema.

El proceso comienza con el acceso del chofer al módulo de agenda, donde selecciona fecha, hora y el camión correspondiente. El sistema responde mostrando el calendario y los horarios disponibles, y posteriormente valida los datos ingresados y la disponibilidad.

Si la validación falla, el sistema muestra un mensaje de error o sugiere horarios alternativos para que el usuario corrija su selección o cancele la operación. En caso de ser correcta, la cita se registra en la base de datos y el sistema muestra la confirmación correspondiente.

Este diagrama evidencia cómo se estructuran las **decisiones, validaciones y posibles excepciones** que ocurren durante el proceso de agendamiento, asegurando una correcta gestión de las citas.



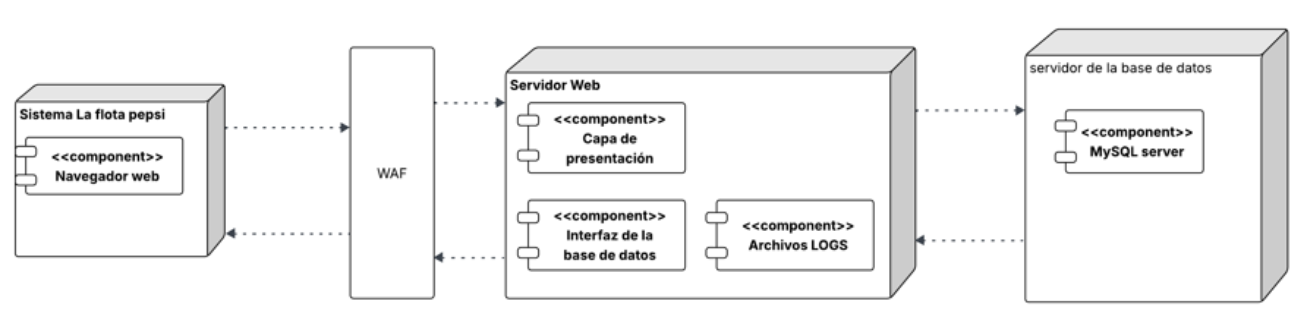
# Diagrama de despliegue

El diagrama de despliegue muestra la arquitectura física y lógica en la que se ejecutará el sistema *La Flota Pepsiman*. Representa los principales nodos y componentes que interactúan durante la operación de la aplicación:

* Cliente (Sistema La Flota Pepsi): accede al sistema a través del navegador web.
* WAF (Web Application Firewall): actúa como capa de seguridad, filtrando las peticiones antes de que lleguen al servidor web.
* Servidor Web: contiene la capa de presentación, la interfaz de la base de datos y los archivos de logs, gestionando la lógica principal de la aplicación.
* Servidor de Base de Datos: aloja MySQL Server, encargado de almacenar de manera persistente la información del sistema.

Las conexiones entre los nodos están representadas mediante protocolos seguros (HTTPS/TLS entre cliente y servidor web, y TCP/IP para la comunicación con la base de datos).

Este diagrama permite visualizar la distribución de los componentes en la infraestructura, evidenciando tanto la seguridad como la organización de los elementos necesarios para la futura implementación en Laravel con MySQL.



# Conclusión

La elaboración de los distintos diagramas permitió estructurar y documentar de manera clara la fase de diseño del sistema *La Flota Pepsiman*. A través de los diagramas de casos de uso, se identificaron los actores y las principales funcionalidades requeridas, asegurando que el sistema cubra las necesidades de clientes, mecánicos, jefes de taller y administradores.

Los diagramas de secuencia detallaron la interacción paso a paso entre los actores y los componentes del sistema, proporcionando una visión dinámica de los procesos clave, como el agendamiento de citas y la actualización de estados de reparación. Asimismo, el diagrama de actividades permitió modelar el flujo de trabajo y las decisiones involucradas en las operaciones, identificando posibles excepciones y alternativas de ejecución.

Finalmente, el diagrama de despliegue mostró la arquitectura física y lógica donde funcionará el sistema, definiendo los nodos principales, los componentes y los protocolos de comunicación, lo cual asegura una base sólida para la futura implementación en Laravel con MySQL.

En conjunto, estos diagramas constituyen una guía fundamental que orientará el desarrollo, las pruebas y la validación del sistema, asegurando coherencia entre los requerimientos, el diseño y la implementación final.