

Documento de Arquitectura Sistema de Monitoreo y Telemetría para Empresa de Buses

Jesús David Mejía Vergara Cod. 30000131476

Christian David Fajardo Alzate Cod. 30000132844

Edgar Gustavo Navarro Renza Cod. 30000130937

Jhon Edinson Acevedo Rojas Cod. 30000118860

Juan Felipe Guevara Grijalba Cod. 300001145761

ARQUITECTURA DE SOFTWARE

UNIVERSIDAD SAN BUENAVENTURA CALI

Control de versiones	3
Introducción	4
Propósito del documento	4



Alcance del proyecto.....	4
Contexto y antecedentes	5
Terminología y definiciones	5
Público objetivo	5
Visión General del Sistema	6
Descripción General del Sistema	6
Objetivos del Sistema.....	7
Beneficios esperados	8
Resumen de Funcionalidades	8
Componentes principales del sistema	9
Diagrama de Alto Nivel	10
Stakeholders.....	11
Requisitos del sistema.....	12
Requisitos no funcionales	13
Restricciones y supuestos	13
Requerimientos de calidad	20
Estilos de Arquitectura y Patrones de Arquitectura	25
Estilos de Arquitectura	26
Arquitectura Basada en Microservicios	26
Arquitectura de Event-Driven (Basada en Eventos)	26
Arquitectura de Tuberías y Filtros (Pipes and Filters Architecture)	26
Arquitectura de Mensajería	27
Patrones de Arquitectura	27
Arquitectura de Contenedores y Orquestación	28
Arquitectura de API Gateway	28
Discusiones de Arquitectura	29
Uso de Kubernetes en Cada Autobús	29
Contenedores Docker para los Sensores y el Colector de Datos	30
Pods para Cada Tipo de Sensor	30
Colas de Mensajería (RabbitMQ o Kafka)	31
Transmisión de Datos en Formato JSON.....	32
API Gateway en la Central.....	33

Microservicios para Procesamiento y Almacenamiento de Datos	33
Microservicio de Monitoreo de Velocidad y Sistema de Alertas.....	34
Almacenamiento de Datos en Bases de Datos	34
Seguridad de los Datos.....	35
Cumplimiento Normativo	35
Diseño de la arquitectura.....	36
Diagrama de Contexto General.....	36
Diagrama de Componentes	37
Diagrama de Despliegue	38
Diagrama de Secuencia.....	40
Diagramas C4	42
Diagrama de contexto.....	43
Diagrama de Contenedores	44
Diagrama de Componentes	45

Control de versiones

Fecha	Versión	Descripción	Autor
20-07-2024	1.0	Creación del documento	Jesus David Mejia Vergara Christian David Fajardo Alzate Edgar Gustavo Navarro Renza Jhon Edinson Acevedo Rojas

Introducción

Propósito del documento

El propósito de este Documento de Arquitectura de Sistemas (DAS) es proporcionar una descripción detallada y comprensiva de la arquitectura del sistema de monitoreo y telemetría para "Buses Seguros S.A.". Este documento sirve como guía completa para diseñar, desarrollar e implementar el sistema, ofreciendo una visión integral de la estructura de los componentes del sistema, los procesos de recopilación y transmisión de datos, y las tecnologías utilizadas para asegurar la confiabilidad y eficiencia del sistema. Además, el documento aborda los requisitos de seguridad, mantenimiento y cumplimiento normativo, facilitando la comprensión y alineación entre los equipos técnicos de ambas empresas y asegurando que se cumplan los requerimientos y expectativas del proyecto.

Alcance del proyecto

- **Implementación de sensores y clusters Kubernetes en cada autobús:** Cada autobús será equipado con sensores para monitorear diversos aspectos (combustible, presión de aire, ambiente, temperatura, velocidad y GPS) y un clúster Kubernetes para gestionar los contenedores que procesan estos datos.
- **Desarrollo e implementación de pods para cada tipo de sensor y un pod colector de datos:** Cada sensor tendrá su propio pod dedicado, y habrá un pod central que recolectará y procesará los datos de todos los sensores.
- **Configuración de colas de mensajería para el almacenamiento temporal de datos:** Se utilizarán colas de mensajería como RabbitMQ o Kafka para asegurar que los datos se almacenen temporalmente y se transmitan de manera confiable a la central.
- **Desarrollo de API Gateway y microservicios en la central para recepción, procesamiento y almacenamiento de datos:** En la central, se implementará un API Gateway para recibir los datos y varios microservicios para procesarlos y almacenarlos en bases de datos.



- **Implementación de un sistema de monitoreo en tiempo real y generación de alertas:** Se desarrollará un microservicio dedicado para monitorear la velocidad de los autobuses y generar alertas automáticas en caso de exceso de velocidad.

Contexto y antecedentes

"Buses Seguros S.A." es una empresa de transporte de pasajeros que opera una flota de autobuses en diversas rutas. La empresa busca mejorar la seguridad y la eficiencia operativa de su flota mediante la implementación de un sistema avanzado de monitoreo y telemetría. Actualmente, la empresa enfrenta desafíos en la gestión y mantenimiento de su flota debido a la falta de información en tiempo real sobre el estado de los autobuses. Implementar un sistema de monitoreo y telemetría permitirá a la empresa obtener datos precisos y actualizados, mejorar la seguridad vial, reducir costos operativos y cumplir con las normativas de transporte.

Terminología y definiciones

- **DAS:** Documento de Arquitectura de Sistemas
- **API:** Interfaz de Programación de Aplicaciones (Application Programming Interface)
- **Kubernetes:** Plataforma de orquestación de contenedores
- **RabbitMQ:** Software de broker de mensajes
- **Kafka:** Plataforma de streaming distribuida
- **JSON:** Notación de Objetos JavaScript (JavaScript Object Notation)
- **Pods:** Unidad de despliegue más pequeña en Kubernetes, que puede contener uno o más contenedores
- **GPS:** Sistema de Posicionamiento Global (Global Positioning System)

Público objetivo

Este documento está dirigido a:



- **Ingenieros de software:** Que participarán en el diseño, desarrollo e implementación del sistema. Este grupo necesita entender la arquitectura técnica y los componentes específicos del sistema para asegurar una implementación exitosa.
- **Gerentes de proyecto:** Que supervisarán el progreso del proyecto y asegurarán que se cumplan los objetivos. Este grupo necesita una comprensión clara de los objetivos del proyecto, los beneficios esperados y los hitos clave para gestionar el proyecto de manera efectiva.
- **Stakeholders y directivos de la empresa:** Que necesitan entender los beneficios y el impacto del sistema en la operación de la flota. Este grupo requiere información detallada sobre cómo el sistema mejorará la seguridad y la eficiencia operativa, así como los beneficios financieros y de cumplimiento regulatorio.
- **Especialistas en seguridad y cumplimiento:** Que garantizarán que el sistema cumpla con las normativas y regulaciones pertinentes. Este grupo necesita asegurarse de que el sistema implemente todas las medidas de seguridad necesarias y cumpla con las regulaciones locales e internacionales.

Visión General del Sistema

Descripción General del Sistema

El sistema de monitoreo y telemetría para "Buses Seguros S.A." es una solución integral diseñada para mejorar la seguridad y eficiencia operativa de la flota de autobuses de la empresa. Cada autobús está equipado con un conjunto de sensores que recolectan datos en tiempo real sobre diversos aspectos del vehículo, como el nivel de combustible, la presión de aire en las llantas, las condiciones ambientales, la temperatura, la velocidad y la localización GPS. Estos datos son procesados localmente en cada autobús utilizando un cluster Kubernetes, que gestiona contenedores Docker para cada tipo de sensor. Los datos recolectados son almacenados temporalmente en colas de mensajería (RabbitMQ o Kafka) y transmitidos a la central de la empresa cuando los autobuses llegan a las terminales.

En la central, un sistema de microservicios procesa y almacena estos datos, permitiendo su análisis y la generación de alertas en tiempo real. Además, el sistema asegura el cumplimiento de regulaciones locales e internacionales relacionadas con la operación de



vehículos de transporte, especialmente en lo que respecta a la velocidad y las condiciones de operación.

Objetivos del Sistema

1. **Monitoreo en Tiempo Real:** Capturar y procesar datos de sensores en tiempo real para proporcionar una visión actualizada de las condiciones y el rendimiento de cada autobús, permitiendo una rápida respuesta ante posibles problemas.
2. **Almacenamiento y Transmisión de Datos Confiable:** Utilizar colas de mensajería como RabbitMQ o Kafka para asegurar que todos los datos de los sensores se almacenen temporalmente y se transmitan de manera confiable a la central, minimizando la pérdida de información.
3. **Escalabilidad y Flexibilidad:** Implementar un cluster Kubernetes en cada autobús para gestionar los contenedores que procesan los datos de los sensores, permitiendo la fácil adición de nuevos sensores o la modificación de los existentes sin interrumpir el sistema.
4. **Integración Eficiente con la Central:** Transmitir los datos recopilados en formato JSON a los servicios de recepción en la central cuando los autobuses arriben a las terminales, asegurando una integración fluida y eficiente con los sistemas centrales.
5. **Análisis y Almacenamiento de Datos:** Procesar y almacenar los datos recibidos en bases de datos en la central para su posterior análisis, permitiendo la generación de informes y la toma de decisiones basadas en datos históricos y en tiempo real.
6. **Generación de Alertas:** Monitorear la velocidad del autobús en tiempo real y generar alertas automáticas si se superan las restricciones locales, mejorando la seguridad y el cumplimiento de las normativas.
7. **Optimización de Recursos:** Utilizar contenedores Docker y clusters Kubernetes para optimizar el uso de recursos en cada autobús, asegurando que el sistema sea eficiente y tenga un bajo impacto en el rendimiento del autobús.
8. **Mantenimiento y Actualización Fácil:** Facilitar el mantenimiento y la actualización del sistema mediante el uso de tecnologías de contenedorización y orquestación.



(Docker y Kubernetes), permitiendo desplegar nuevas versiones de software sin interrupciones significativas.

9. **Seguridad de los Datos:** Asegurar que los datos transmitidos desde los autobuses a la central estén protegidos mediante encriptación y mecanismos de autenticación, garantizando la privacidad y la integridad de la información.
10. **Cumplimiento Normativo:** Cumplir con las regulaciones locales e internacionales respecto a la operación de vehículos de transporte, especialmente en lo que se refiere a la velocidad y las condiciones de operación, ayudando a la empresa a evitar multas y mejorar la seguridad en las carreteras.

Beneficios esperados

- **Mejora en la seguridad y eficiencia:** El monitoreo constante y en tiempo real de las condiciones de los autobuses permitirá prevenir incidentes y optimizar el uso de recursos, asegurando que los autobuses operen en condiciones óptimas y seguras.
- **Reducción de costos operativos:** Al obtener datos precisos y en tiempo real sobre el estado de los autobuses, se podrá planificar mejor el mantenimiento y evitar reparaciones costosas no planificadas. Esto también reducirá el tiempo de inactividad de los autobuses.
- **Cumplimiento regulatorio:** El sistema ayudará a la empresa a cumplir con las normativas de seguridad y operación de los vehículos, evitando multas y sanciones. Las alertas automáticas en caso de exceso de velocidad contribuirán a mejorar la seguridad vial.
- **Mejora en la toma de decisiones:** La capacidad de almacenar y analizar datos históricos y en tiempo real permitirá a la empresa tomar decisiones informadas y estratégicas, basadas en un análisis exhaustivo de la información recopilada.
- **Optimización de la flota:** La información precisa y en tiempo real sobre el estado de cada autobús permitirá una mejor gestión de la flota, optimizando las rutas y mejorando la eficiencia operativa.

Resumen de Funcionalidades

El sistema proporciona las siguientes funcionalidades clave:



1. **Recolección de Datos en Tiempo Real:** Captura de datos de sensores como nivel de combustible, presión de aire, condiciones ambientales, temperatura, velocidad y GPS.
2. **Procesamiento de Datos:** Uso de clusters Kubernetes para procesar datos de sensores en cada autobús.
3. **Almacenamiento Temporal de Datos:** Utilización de colas de mensajería (RabbitMQ o Kafka) para asegurar la disponibilidad de los datos antes de su transmisión.
4. **Transmisión de Datos a la Central:** Envío de datos en formato JSON a la central de la empresa para análisis y almacenamiento.
5. **Generación de Alertas en Tiempo Real:** Monitoreo y notificación de eventos críticos, como exceso de velocidad.
6. **Integración con Sistemas Centrales:** Uso de API Gateway y microservicios para gestionar la recepción y procesamiento de datos en la central.
7. **Seguridad de Datos:** Implementación de encriptación y autenticación para proteger la integridad y privacidad de los datos.

Componentes principales del sistema

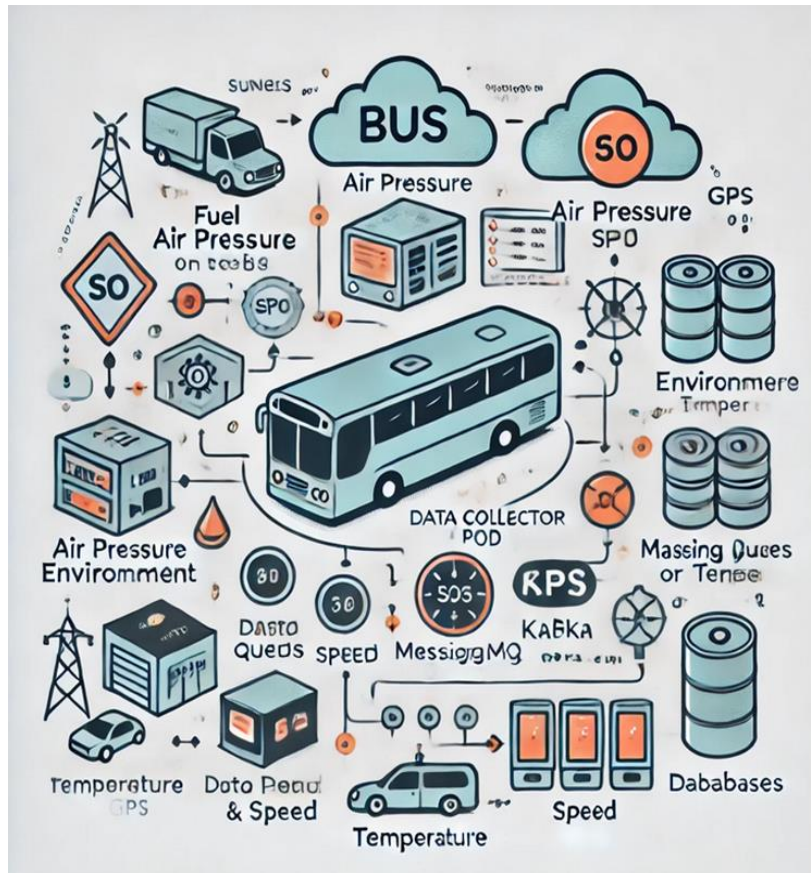
El sistema incluye los siguientes componentes principales:

- **Sensores:** Dispositivos físicos instalados en cada autobús que recopilan datos específicos (combustible, presión de aire, ambiente, temperatura, velocidad y GPS).
- **Cluster Kubernetes en el Autobús:** Un cluster Kubernetes que ejecuta pods Docker para cada sensor, así como un pod colector de datos que centraliza la información de los sensores.
- **Colas de Mensajería:** Utilizadas para almacenar temporalmente los datos antes de su transmisión a la central.
- **API Gateway:** En la central, recibe las solicitudes de datos desde los autobuses.
- **Microservicios:** Procesan y almacenan los datos recibidos en bases de datos en la central.

- **Sistema de Monitoreo y Alertas:** Monitorea la velocidad del autobús en tiempo real y genera alertas automáticas si se superan las restricciones locales.

Diagrama de Alto Nivel

El siguiente diagrama de alto nivel ilustra la arquitectura propuesta para el sistema de monitoreo y telemetría de "Buses Seguros S.A.". Este diagrama destaca los componentes principales y su interacción desde la recolección de datos en el autobús hasta su procesamiento y almacenamiento en la central. La arquitectura está diseñada para garantizar la recolección confiable de datos, su procesamiento en tiempo real, y la generación de alertas automáticas para mejorar la seguridad y eficiencia operativa de la flota de autobuses. Los componentes clave incluyen sensores en el autobús, un cluster Kubernetes para la gestión de contenedores, colas de mensajería para el almacenamiento temporal de datos, un API Gateway para la recepción de datos, microservicios para el procesamiento y almacenamiento de información, y un sistema de monitoreo y alertas en tiempo real.



Stakeholders

Stakeholders	Descripción	Expectativas
Equipo de TI	Responsable de la infraestructura tecnológica y del soporte técnico.	Sistema confiable y fácil de mantener, con alta disponibilidad y un rendimiento óptimo.
Equipo de Seguridad de TI	Encargado de la seguridad de la información y de las comunicaciones.	Datos protegidos mediante encriptación y autenticación, sin brechas de seguridad.
Equipo de Desarrollo de Software	Desarrolladores y arquitectos de software encargados de la implementación del sistema.	Código eficiente, escalable y fácil de actualizar, con mínima interrupción del servicio durante despliegues.



Equipo de Operaciones	Monitorea la operación diaria de la flota de autobuses.	Sistema de monitoreo en tiempo real preciso y confiable, con generación rápida de alertas.
Equipo de Mantenimiento	Responsable del mantenimiento del sistema y de los autobuses.	Procedimientos de mantenimiento y actualización del software que no interfieran con la operación de los autobuses.
Departamento Legal y de Cumplimiento	Asegura que la empresa cumpla con todas las regulaciones locales e internacionales.	Cumplimiento de todas las normativas, especialmente en cuanto a velocidad y condiciones de operación de los autobuses.
Equipo de Análisis de Datos	Analistas encargados de procesar y analizar los datos recolectados.	Acceso a datos históricos y en tiempo real de manera confiable para la generación de informes y toma de decisiones.
Operadores de la Central	Personal que gestiona el sistema de recepción y monitoreo de datos en la central.	Interfaz de usuario intuitiva y eficiente, que permita responder rápidamente a las alertas y gestionar datos recibidos.
Clientes/Pasajeros	Usuarios finales que utilizan el servicio de transporte.	Mayor seguridad y eficiencia en el servicio de transporte, con una experiencia de viaje cómoda y sin interrupciones.
Alta Gerencia	Directivos responsables de la toma de decisiones estratégicas en la empresa.	Sistema que mejore la seguridad y eficiencia operativa, proporcionando datos fiables para la toma de decisiones estratégicas.

Requisitos del sistema



Requisitos no funcionales

Requisitos No Funcionales		
#	Nombre	Items
1	Rendimiento	<ul style="list-style-type: none">• Procesamiento de datos en tiempo real con baja latencia.• Transmisión confiable de datos de los sensores relevantes en tiempo real como porcentaje restante de combustible, nivel de sueño del conductor, velocidad, temperatura y nivel de aire en las llantas.• Garantizar la rápida respuesta y alerta oportuna de los sensores mencionados en el ítem anterior.
2	Escalabilidad	<ul style="list-style-type: none">• Implementar un cluster Kubernetes en cada autobús para gestionar los contenedores.• Facilitar la adición de nuevos sensores o la modificación de los existentes sin interrumpir el sistema.• Filtrar los datos de los sensores primarios en las colas de mensajería, para optimizar la transferencia de datos en tiempo real.
3	Seguridad	<ul style="list-style-type: none">• Proteger los datos transmitidos desde los autobuses a la central mediante encriptación y mecanismos de autenticación.• Garantizar la privacidad y la integridad de la información.• Descargar todos los datos represados en la cola de mensajería, solo cuando se llegue a una central con una conexión segura a internet.
4	Mantenimiento y Actualización	<ul style="list-style-type: none">• Facilitar el mantenimiento y la actualización del sistema mediante el uso de tecnologías de contenerización y orquestación (Docker y Kubernetes).• Desplegar nuevas versiones del software de manera rápida y segura. Solo cuando los autobuses se reporten en alguna de las centrales.

Restricciones y supuestos



ID Restricción R01	Tipo Tecnología (X) Negocio ()	Nombre Precisión y Confiabilidad de los Sensores
Descripción:	Los sensores de combustible, presión de aire en las llantas, ambiente, temperatura, velocidad y GPS deben ser precisos y confiables.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Utilizar sensores de diferentes proveedores para comparar su precisión y confiabilidad.	
Observaciones:	La precisión de los sensores impacta directamente en la calidad de los datos recolectados.	

ID Restricción R02	Tipo Tecnología (X) Negocio ()	Nombre Capacidad de Hardware del Autobús
Descripción:	El hardware de cada bus debe ser capaz de ejecutar un cluster Kubernetes con los recursos necesarios (CPU, memoria, almacenamiento).	
Establecida por:	Equipo de TI de Buses Seguros S.A.	
Alternativas:	Evaluar y seleccionar hardware especializado que soporte Kubernetes en entornos limitados.	
Observaciones:	La selección de hardware adecuado es crucial para el rendimiento del sistema.	

ID Restricción R03	Tipo Tecnología (X) Negocio ()	Nombre Formato de Transmisión de Datos
Descripción:	Los datos deben ser transmitidos en formato JSON para asegurar compatibilidad con los servicios de recepción.	
Establecida por:	Equipo de Desarrollo de Software.	
Alternativas:	Utilizar otros formatos de transmisión (e.g., XML) si se justifican técnicamente.	



Observaciones:	JSON es un formato ampliamente soportado y eficiente para la transmisión de datos.
-----------------------	--

ID Restricción R04	Tipo Tecnología (X) Negocio ()	Nombre Uso de Colas de Mensajería
Descripción:	Uso de colas de mensajería (RabbitMQ o Kafka) para asegurar que no haya pérdida de datos durante la transmisión.	
Establecida por:	Equipo de Arquitectura de Sistemas.	
Alternativas:	Evaluar otras soluciones de mensajería (e.g., MQTT) si se ajustan mejor a los requisitos.	
Observaciones:	La elección de la tecnología de mensajería es fundamental para la integridad de los datos.	

ID Restricción R05	Tipo Tecnología (X) Negocio ()	Nombre Seguridad de los Datos
Descripción:	Los datos transmitidos desde los autobuses a la central deben estar encriptados para garantizar su privacidad.	
Establecida por:	Equipo de Seguridad de TI.	
Alternativas:	Utilizar diferentes protocolos de encriptación y autenticación.	
Observaciones:	La encriptación protege los datos contra accesos no autorizados.	

ID Restricción R06	Tipo Tecnología (X) Negocio ()	Nombre Autenticación de Transmisiones de Datos
Descripción:	Implementación de mecanismos de autenticación para asegurar que solo fuentes autorizadas puedan transmitir datos.	



Establecida por:	Equipo de Seguridad de TI.
Alternativas:	Uso de certificados digitales y autenticación de dos factores.
Observaciones:	La autenticación asegura que los datos sean recibidos solo de fuentes confiables.

ID Restricción R07	Tipo Tecnología (X) Negocio ()	Nombre Escalabilidad del Sistema en el Autobús
Descripción:	El cluster Kubernetes debe ser capaz de manejar la adición de nuevos sensores o modificación de los existentes sin interrumpir el servicio.	
Establecida por:	Equipo de Arquitectura de Sistemas.	
Alternativas:	Diseñar una arquitectura modular que permita la fácil integración de nuevos componentes.	
Observaciones:	La escalabilidad es crucial para la evolución del sistema.	

ID Restricción R08	Tipo Tecnología (X) Negocio ()	Nombre Escalabilidad del Sistema en la Central
Descripción:	La infraestructura en la central debe ser capaz de procesar y almacenar grandes volúmenes de datos recibidos de múltiples autobuses.	
Establecida por:	Equipo de Arquitectura de Sistemas.	
Alternativas:	Implementar soluciones de escalado horizontal y vertical para la infraestructura central.	
Observaciones:	La capacidad de escalar es esencial para manejar el crecimiento del sistema.	

ID Restricción R09	Tipo Tecnología (X) Negocio ()	Nombre Mantenimiento del Sistema en el Autobús
-------------------------------	--	---



Descripción:	Facilitar el mantenimiento y la actualización del sistema mediante el uso de tecnologías de contenedorización y orquestación.
Establecida por:	Equipo de TI de Buses Seguros S.A.
Alternativas:	Utilizar herramientas de automatización de despliegue y gestión de configuraciones.
Observaciones:	La facilidad de mantenimiento reduce los tiempos de inactividad.

ID Restricción R10	Tipo Tecnología (X) Negocio ()	Nombre Mantenimiento del Sistema en la Central
Descripción:	Los microservicios deben ser diseñados para permitir actualizaciones y mantenimientos sin afectar la operación continua del sistema.	
Establecida por:	Equipo de Desarrollo de Software	
Alternativas:	Implementar microservicios con arquitectura desacoplada y CI/CD.	
Observaciones:	Un buen diseño de microservicios facilita el mantenimiento y actualización.	

ID Restricción R11	Tipo Tecnología (X) Negocio ()	Nombre Procesamiento en Tiempo Real
Descripción:	Los datos deben ser procesados en tiempo real para proporcionar una visión actualizada de las condiciones y el rendimiento del autobús.	
Establecida por:	Equipo de Arquitectura de Sistemas	
Alternativas:	Uso de tecnologías de procesamiento en tiempo real como Apache Flink o Spark Streaming	
Observaciones:	El procesamiento en tiempo real es clave para la generación oportuna de alertas.	



ID Restricción R12	Tipo Tecnología (X) Negocio ()	Nombre Almacenamiento y Recuperación de Datos
Descripción:	Los datos deben ser almacenados de manera eficiente y ser fácilmente accesibles para análisis posteriores.	
Establecida por:	Equipo de Desarrollo de Software	
Alternativas:	Uso de bases de datos NoSQL y soluciones de almacenamiento distribuido.	
Observaciones:	Un almacenamiento eficiente mejora la capacidad de análisis de datos.	

ID Restricción R13	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	Asegurar el cumplimiento de todas las regulaciones locales e internacionales relacionadas con la operación de vehículos de transporte.	
Establecida por:	Departamento Legal y de Cumplimiento	
Alternativas:	Monitorear cambios en las regulaciones y ajustar el sistema según sea necesario	
Observaciones:	El cumplimiento normativo es esencial para evitar sanciones y mejorar la seguridad.	

ID Restricción R14	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	La unidad de monitoreo de monitoreo soportará hasta 8 sensores conectados.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Desarrollar unidades con capacidad para más puertos para conectar sensores.	



Observaciones:	La capacidad de la unidad de monitoreo solo permite 8 puertos a los cuales se pueden conectar los diferentes sensores.
-----------------------	--

ID Restricción R15	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	Los sensores que se pueden conectar a la unidad de monitoreo son los siguientes: <ul style="list-style-type: none">– Sensor de encendido y apagado de vehículo– Sensor de presión de aire en llantas– Sensor de velocidad– Sensor de combustible– Sensor de velocidad– Sensor de temperatura interna– Sensor de temperatura ambiente– Cámara en cabina	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Determinar los sensores más importantes o integrar otra unidad al vehículo. Desarrollar unidades con capacidad para más puertos para conectar sensores.	
Observaciones:	Dadas las limitaciones físicas de las unidades, solo se pueden conectar hasta 8 sensores por unidad.	

ID Restricción R16	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	Las unidades de monitoreo se deben desconectar de la batería del vehículo cuando este no sea utilizado por un periodo prolongado para no agotar la batería.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Utilizar unidades con baterías portátiles. Se debe cambiar la batería cada que se agote.	



Observaciones:	La unidad de monitoreo se conecta y es alimentada por la batería del vehículo, la cual se carga cuando el vehículo está encendido, si el vehículo es apagado por un tiempo prolongado, la unidad seguirá operando y consumiendo la batería del vehículo hasta que esta se agote.
-----------------------	--

ID Restricción R17	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	La unidad de monitoreo contará con una interfaz para descarga de los datos vía bluetooth.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Descargar los datos a través de un puerto USB.	
Observaciones:	Dependiendo de la forma que se descarguen los datos, se utilizará un módulo bluetooth o un puerto USB, por lo que se debe configurar de forma diferente cada unidad. Es posible configurar ambos métodos de descarga, pero se deberá utilizar un puerto usado por un sensor.	

ID Restricción R18	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	En caso de baja o nula cobertura, las alarmas se generarán de forma extemporánea.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Implementar sistemas SMS, pero este no tiene cobertura completa.	
Observaciones:	Dadas las condiciones geográficas y la cobertura de las redes, hay lugares donde los dispositivos se les dificultará o imposibilitará la transmisión de las alertas, por lo cual estas serán enviadas de forma extemporánea o no se generarán según configuración. En todo caso, estas serán almacenadas.	

--	--	--



ID Restricción R19	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	Solo se podrán generar las siguientes alarmas serán notificadas en tiempo real a la central: <ul style="list-style-type: none">– Exceso de velocidad– Perdida de presión de aire en llantas– Bajo nivel de combustible– Micro sueño	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Dotar de mayores recursos las unidades de monitoreo.	
Observaciones:	Dado los recursos de la unidad de monitoreo, solo se analizarán aquellos eventos encaminados a cumplir las normas viales y aquellos eventos que buscan salvaguardar la vida del conductor y los pasajeros.	

ID Restricción R20	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	Los datos de la unidad serán borrados una vez que estos sean descargados.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Dotar las unidades de monitoreo de mayores recursos de almacenamiento.	
Observaciones:	Dado que la capacidad de almacenamiento de la unidad de monitoreo es limitada, solo se persistirán los datos hasta que estos sean descargados por la central y se almacenará máximo hasta 5 días de información, pasado este tiempo los datos será borrados según la unidad requiera el espacio.	

ID Restricción R21	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	Solo se garantizará protección IP63 para las unidades de monitoreo.	



Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.
Alternativas:	Mejorar el sellado o comprar mejores contenedores para las unidades de monitoreo.
Observaciones:	La caja de la unidad de monitoreo se encuentra sellada y ofrece una protección IP63, la cual la protegerá del polvo y la exposición a chorros de agua, pero esta no soportará ser sumergida completamente bajo el agua.

ID Restricción R22	Tipo Tecnología (X) Negocio ()	Nombre Cumplimiento con Regulaciones
Descripción:	La cámara de monitoreo de cabina solo generará alarma de micro sueño notificando el evento, mas no habrá transmisión de imágenes.	
Establecida por:	Equipo de Ingeniería de Buses Seguros S.A.	
Alternativas:	Mejorar el módulo de transmisión de datos para que realice el envío de imágenes y permita el streaming.	
Observaciones:	La cámara de monitoreo de cabina no cuenta con una unidad con capacidad de transmisión de imágenes ni de almacenamiento, solo tiene capacidad de envío de eventos vía mensajes de texto, por lo cual solo se generarán las alertas de texto.	

Atributos de Calidad

Atributo de Calidad:		Eficiencia	
Tiempo	ID	Descripción	Prioridad
Se deben manejar tiempos de envío óptimos para el procesamiento	001	Recolección de datos en tiempo real de todos los sensores.	Alta



adecuado de las alertas.			
Tiempos de procesamiento óptimos para procesar la data importante y enviarla.	002	Filtrado de datos de los sensores para enviar solo los provenientes de los primarios para procesar las alertas.	Alta
Tiempos de procesamiento normales, solo se da prioridad a los datos de los sensores primarios.	003	Encolamiento de datos de sensores no primarios para su descarga en la central. Para las colas de mensajería se debe de utilizar Kafka.	Media
Tiempo de procesamiento normales, ya que solo se hace al llegar a una central de autobuses.	004	Conversión de datos en formato JSON para transmitir la información a la central.	Media
Tiempo de procesamiento optimo, debido a que se deben de vaciar las colas con los datos de los demás sensores al momento de llegar a la central de manera instantánea.	005	Transmisión de datos al momento de llegar a la central.	Alta
Recursos	ID	Descripción	Prioridad
Tiempo de respuesta	1	Medición del tiempo que tarda el sistema en responder a una solicitud.	Alta
Utilización de recursos	2	Evaluación de cómo el sistema utiliza los recursos disponibles (CPU, memoria).	Media
Capacidad	3	Capacidad del sistema para manejar grandes volúmenes de datos y usuarios.	Alta
Volumen de carga	4	Evaluación del rendimiento bajo condiciones de alta carga y estrés.	Media

Atributo de Calidad:		Fiabilidad	
Tolerancia a Fallas	ID	Descripción	Prioridad



Recuperabilidad	001	Capacidad del sistema para recuperarse de fallas y continuar operando.	Alta
Disponibilidad	002	Capacidad del sistema para estar operativo y accesible cuando se necesita.	Media
Redundancia	003	Implementación de componentes de respaldo que pueden tomar el lugar de los componentes fallidos.	Alta

Atributo de Calidad:		Mantenimiento	
Modificación	ID	Descripción	Prioridad
Facilidad de reparación	1	Capacidad del sistema para ser reparado rápidamente y con facilidad.	Alta
Disponibilidad de repuestos	2	Disponibilidad de piezas y componentes necesarios para el mantenimiento.	Media
Documentación técnica	3	Calidad y accesibilidad de la documentación técnica para el mantenimiento.	Alta
Capacitación del personal	4	Nivel de formación y capacitación del personal encargado del mantenimiento.	Media
Facilidad de Pruebas	ID	Descripción	Prioridad
Modularidad	001	La capacidad del sistema para dividirse en módulos independientes.	Alta
Reusabilidad	002	La capacidad de utilizar componentes del sistema en otros contextos.	Media
Analizabilidad	003	La facilidad con la que se pueden identificar las causas de los defectos.	Alta
Modificabilidad	004	La facilidad con la que se pueden realizar cambios en el sistema.	Alta
Estabilidad	005	La capacidad del sistema para evitar efectos adversos debido a cambios.	Media
Testabilidad	006	La facilidad con la que se pueden realizar pruebas al sistema.	Alta



Estabilidad	ID	Descripción	Prioridad
Alta	001	Control de variación de materiales	Alta
Media	002	Eliminación de problemas en el proceso	Media
Alta	003	Capacidad del personal de mantenimiento	Alta
Baja	004	Reducción de errores y desperdicios	Baja
Flexibilidad	ID	Descripción	Prioridad
Alta	001	Capacidad de adaptarse a cambios en los requisitos de mantenimiento.	Alta
Media	002	Implementación de mejoras continuas en los procesos de mantenimiento.	Media
Baja	003	Estandarización de procedimientos para minimizar errores.	Baja
Escalabilidad	ID	Descripción	Prioridad
Alta	001	Capacidad del sistema para manejar un aumento en la carga de trabajo sin afectar el rendimiento.	Alta
Media	002	Adaptabilidad del sistema para incorporar nuevas funcionalidades sin necesidad de rediseño significativo.	Media
Alta	003	Facilidad de mantenimiento y actualización del sistema con mínimo tiempo de inactividad.	Alta
Baja	004	Capacidad del sistema para integrarse con otros sistemas y tecnologías existentes.	Baja

Estilos de Arquitectura y Patrones de Arquitectura

Estilos de Arquitectura

En el diseño del sistema de monitoreo y telemetría para "Buses Seguros S.A.", se han aplicado varios estilos de arquitectura para abordar los diferentes requerimientos del sistema. A continuación, se describen los principales estilos de arquitectura empleados:

Arquitectura Basada en Microservicios

La arquitectura de microservicios divide la aplicación en múltiples servicios independientes, cada uno con una funcionalidad específica y autónoma. En este sistema, los microservicios se utilizan para:

- **Procesamiento de Datos:** Servicios específicos para manejar los datos de los diferentes sensores (combustible, presión de aire, ambiente, temperatura, velocidad, GPS).
- **Monitoreo y Alerta:** Servicios dedicados al monitoreo de condiciones críticas como la velocidad y la generación de alertas en tiempo real.
- **Almacenamiento y Análisis:** Servicios que gestionan el almacenamiento de datos y facilitan el análisis histórico y en tiempo real.

Este estilo permite una alta flexibilidad y escalabilidad, facilitando la implementación de nuevas funcionalidades y la actualización de componentes individuales sin afectar el sistema en su totalidad.

Arquitectura de Event-Driven (Basada en Eventos)

El sistema utiliza un enfoque de arquitectura basada en eventos para manejar la comunicación entre los componentes. Los eventos generados por los sensores (por ejemplo, un cambio en la velocidad del autobús) son capturados y procesados por los microservicios correspondientes. Este patrón es útil para implementar el monitoreo en tiempo real y la generación de alertas.

Las colas de mensajería, como RabbitMQ o Kafka, juegan un papel crucial en este estilo de arquitectura, actuando como mediadores para asegurar la entrega de eventos y manejar el flujo de datos de manera asíncrona.

Arquitectura de Tuberías y Filtros (Pipes and Filters Architecture)

Este estilo es útil para el procesamiento de datos en serie, donde los datos pasan a través de una secuencia de filtros (procesos) para ser transformados. En este proyecto, puede

aplicarse para procesar los datos de los sensores antes de almacenarlos o analizarlos, asegurando que solo la información relevante y filtrada llegue a los sistemas centrales.

Arquitectura de Mensajería

El uso de colas de mensajería (como RabbitMQ o Kafka) en el sistema asegura que los datos de los sensores se almacenen y se procesen de manera confiable. Este enfoque proporciona:

- **Desacoplamiento:** Permite que los productores de datos (sensores) y los consumidores de datos (servicios de la central) funcionen de forma independiente.
- **Tolerancia a Fallos:** Las colas de mensajería pueden almacenar mensajes temporalmente si uno de los componentes no está disponible, evitando la pérdida de datos.
- **Escalabilidad:** Facilita la escalabilidad horizontal, permitiendo agregar más consumidores de datos según sea necesario.

Patrones de Arquitectura

Además de los estilos de arquitectura, se han utilizado varios patrones de diseño para estructurar y organizar los componentes del sistema. Los siguientes patrones destacan en este contexto:



Arquitectura de Contenedores y Orquestación

Descripción: Este patrón utiliza contenedores (por ejemplo, Docker) para empaquetar aplicaciones y sus dependencias, y herramientas de orquestación (por ejemplo, Kubernetes) para gestionar el despliegue, escalado y operación de estos contenedores.

Aplicación en el Proyecto:

- **Cluster Kubernetes en Cada Autobús:** Permite gestionar los pods que ejecutan contenedores para cada tipo de sensor y el colector de datos, facilitando la administración y escalabilidad.
- **Contenedores Docker:** Empaquetan aplicaciones para la recolección, procesamiento y transmisión de datos.

Ventajas:

- **Portabilidad:** Facilita el despliegue en diferentes entornos (producción, pruebas, desarrollo).
- **Escalabilidad:** Permite escalar aplicaciones y servicios fácilmente.
- **Mantenimiento y Actualización:** Facilita la actualización y mantenimiento del sistema sin interrumpir el servicio.

Arquitectura de API Gateway

Descripción: Un API Gateway actúa como un punto de entrada único para las solicitudes de clientes, gestionando la autenticación, enrutamiento, y agregación de respuestas.

Aplicación en el Proyecto:

- **Recepción de Datos en la Central:** El API Gateway recibe las solicitudes de datos desde los autobuses, realiza la autenticación y enrutamiento adecuado a los microservicios correspondientes.

Ventajas:

- **Centralización:** Proporciona un punto centralizado para gestionar las solicitudes y la seguridad.
- **Facilita la Escalabilidad:** Permite una fácil integración de nuevos servicios y endpoints

Discusiones de Arquitectura

En el diseño del sistema de monitoreo y telemetría para la empresa de transporte "Buses Seguros S.A.", se tomaron decisiones arquitectónicas clave para asegurar la eficiencia, escalabilidad, fiabilidad y seguridad del sistema. Cada decisión se ha considerado cuidadosamente para cumplir con los objetivos específicos del proyecto y enfrentar los desafíos de la operación de una flota de autobuses en tiempo real. A continuación, se detallan las decisiones arquitectónicas más relevantes, junto con sus respectivas justificaciones:

Uso de Kubernetes en Cada Autobús

Descripción: Cada autobús cuenta con un cluster Kubernetes para gestionar y orquestar los contenedores que ejecutan las aplicaciones de monitoreo y procesamiento de datos. Este cluster incluye varios pods, que son las unidades básicas de ejecución en Kubernetes, y gestionan la ejecución de contenedores Docker.

Justificación:

- **Escalabilidad y Flexibilidad:** Kubernetes permite gestionar múltiples instancias de aplicaciones (contenedores) en un solo nodo. Esto facilita la escalabilidad horizontal, permitiendo añadir más contenedores (por ejemplo, para nuevos tipos de sensores) sin necesidad de reconfigurar el sistema completo. Además, Kubernetes maneja la redistribución de contenedores en caso de fallos, asegurando que el sistema se mantenga operativo.
- **Orquestación y Gestión de Recursos:** Kubernetes proporciona un sistema de gestión de recursos eficiente, donde los contenedores se distribuyen según la carga y los recursos disponibles, evitando sobrecargas y asegurando un uso óptimo de los recursos del autobús. Esto es crucial para mantener el rendimiento del autobús mientras se ejecutan las aplicaciones de monitoreo.
- **Automatización de Despliegues:** Kubernetes soporta actualizaciones continuas y despliegues automatizados, lo que facilita



la implementación de nuevas versiones de software sin interrumpir el funcionamiento del sistema.

Contenedores Docker para los Sensores y el Colector de Datos

Descripción: Se utilizan contenedores Docker para ejecutar las aplicaciones que procesan los datos de los sensores y el pod colector de datos. Docker proporciona un entorno de ejecución estandarizado para estas aplicaciones.

Justificación:

- **Aislamiento y Portabilidad:** Docker asegura que las aplicaciones se ejecuten de manera aislada, evitando conflictos entre diferentes aplicaciones o entre aplicaciones y el sistema operativo del autobús. Esto también facilita la portabilidad de las aplicaciones, permitiendo que se ejecuten en diferentes entornos (desarrollo, prueba, producción) sin necesidad de modificar el código.
- **Facilidad de Despliegue:** Docker simplifica el proceso de despliegue de aplicaciones, ya que los contenedores incluyen todas las dependencias necesarias para ejecutar la aplicación. Esto reduce el tiempo y el esfuerzo necesarios para desplegar y actualizar las aplicaciones en el autobús.

Pods para Cada Tipo de Sensor

Descripción: Se asigna un pod separado para cada tipo de sensor en el autobús (combustible, presión de aire, ambiente, temperatura, velocidad, GPS). Cada pod está encargado de la recolección y procesamiento de datos específicos de su sensor.

Justificación:



- **Modularidad:** La separación en pods permite que cada tipo de sensor funcione de manera independiente. Esto facilita el mantenimiento y la actualización del sistema, ya que las modificaciones en un pod no afectan a los demás. También permite una gestión más eficiente de los recursos, ya que cada pod puede ser escalado de manera independiente según la demanda.
- **Optimización del Rendimiento:** La carga de trabajo está distribuida entre múltiples pods, lo que evita cuellos de botella en el procesamiento de datos. Cada pod puede centrarse en procesar los datos de un tipo específico de sensor, lo que mejora la eficiencia del sistema en general.

Colas de Mensajería (RabbitMQ o Kafka)

Descripción: En el diseño del sistema de monitoreo y telemetría para "Buses Seguros S.A.", se decidió utilizar Apache Kafka como la solución de mensajería en lugar de RabbitMQ. Kafka es una plataforma de streaming distribuida que permite la publicación, suscripción, almacenamiento y procesamiento de flujos de datos en tiempo real.

Justificación

- **Escalabilidad y Rendimiento:** Kafka es conocido por su alta escalabilidad y capacidad para manejar grandes volúmenes de datos con baja latencia. Dado que cada autobús de la flota generará una gran cantidad de datos de múltiples sensores (combustible, presión de aire, ambiente, temperatura, velocidad y GPS), Kafka es una opción superior debido a su capacidad para escalar horizontalmente y manejar millones de eventos por segundo.
- **Persistencia de Datos:** Kafka almacena los mensajes en un log distribuido y permite la persistencia de datos a largo plazo. Esto es crucial para "Buses Seguros S.A.", ya que los datos de telemetría deben estar disponibles no solo en tiempo real, sino también para análisis históricos y auditorías. RabbitMQ, aunque fiable, no está diseñado para almacenamiento de datos a largo plazo.
- **Procesamiento en Tiempo Real:** Kafka es altamente eficaz en el procesamiento de flujos de datos en tiempo real. Con la necesidad de



generar alertas instantáneas sobre problemas como la velocidad excesiva, Kafka garantiza la entrega de mensajes con baja latencia y alta disponibilidad, lo cual es crítico para la seguridad de los pasajeros.

- **Integración y Ecosistema:** Kafka ofrece un ecosistema robusto con herramientas como Kafka Streams para el procesamiento de flujos en tiempo real y Kafka Connect para la integración con otros sistemas y bases de datos. Este ecosistema proporciona una flexibilidad adicional que facilita la implementación de nuevas funcionalidades y la integración con la infraestructura existente de la empresa.
- **Tolerancia a Fallos y Recuperación:** La arquitectura distribuida de Kafka permite una alta tolerancia a fallos y recuperación automática. Dado que los autobuses de la flota pueden operar en áreas con conectividad intermitente, es crucial que el sistema de mensajería pueda recuperar y retransmitir mensajes sin pérdida de datos cuando se restablezca la conexión.

Transmisión de Datos en Formato JSON

Descripción: Los datos recopilados en el autobús se transmiten en formato JSON cuando el autobús llega a la terminal. JSON (JavaScript Object Notation) es un formato ligero y fácil de leer.

Justificación:

- **Estandarización y Facilidad de Integración:** JSON es un formato ampliamente utilizado en la transmisión de datos y es compatible con una gran cantidad de herramientas y servicios. Utilizar JSON facilita la integración con otros sistemas y servicios en la central, ya que es un formato estandarizado que muchos sistemas pueden procesar fácilmente.
- **Legibilidad y Manipulación de Datos:** JSON es fácil de leer y entender tanto para humanos como para sistemas automatizados. Esto facilita la depuración y el análisis de datos, así como la generación de informes y la toma de decisiones.



API Gateway en la Central

Descripción: Un API Gateway en la central recibe las solicitudes de datos desde los autobuses y actúa como el punto de entrada para todas las solicitudes de datos.

Justificación:

- **Gestión Centralizada de API:** El API Gateway centraliza la gestión de las solicitudes de datos, facilitando el enrutamiento, la autenticación, y la autorización. Esto permite una administración más eficiente de las solicitudes y facilita la implementación de políticas de seguridad y control de acceso.
- **Escalabilidad:** El API Gateway puede manejar grandes volúmenes de tráfico de datos y distribuir las solicitudes a los servicios adecuados en la central. Esto permite escalar el sistema para manejar un número creciente de autobuses sin comprometer el rendimiento.

Microservicios para Procesamiento y Almacenamiento de Datos

Descripción: En la central, se utilizan microservicios para procesar y almacenar los datos recibidos de los autobuses. Cada microservicio tiene una responsabilidad específica, como almacenamiento, análisis, o generación de informes.

Justificación:

- **Desacoplamiento:** La arquitectura de microservicios permite que diferentes componentes del sistema se desarrollen, desplieguen y escalen de manera independiente. Esto facilita la evolución y el mantenimiento del sistema, ya que los cambios en un microservicio no afectan a los demás.
- **Mantenimiento y Actualización:** Los microservicios pueden ser actualizados o reemplazados de manera independiente, lo que simplifica el proceso de mantenimiento y permite la implementación

de nuevas funcionalidades sin interrumpir el funcionamiento del sistema.

Microservicio de Monitoreo de Velocidad y Sistema de Alertas

Descripción: Un microservicio específico se encarga del monitoreo en tiempo real de la velocidad del autobús. Este microservicio verifica los datos de velocidad y genera alertas si se superan las restricciones locales.

Justificación:

- **Monitoreo en Tiempo Real:** El microservicio de monitoreo permite la supervisión continua de la velocidad del autobús, lo que es crucial para detectar y responder rápidamente a violaciones de las restricciones de velocidad. Esto mejora la seguridad del autobús y asegura el cumplimiento de las normativas.
- **Automatización de Alertas:** El sistema de alertas automatiza la notificación de problemas, lo que permite a los operadores actuar rápidamente en caso de exceso de velocidad. Esto reduce el tiempo de respuesta y mejora la eficiencia operativa.

Almacenamiento de Datos en Bases de Datos

Descripción: Los datos transmitidos desde los autobuses se almacenan en bases de datos en la central para su posterior análisis y generación de informes.

Justificación:

- **Análisis y Generación de Informes:** El almacenamiento de datos permite realizar análisis a largo plazo y generar informes detallados sobre el rendimiento de la flota. Esto ayuda a identificar patrones,



prever mantenimientos y tomar decisiones informadas basadas en datos históricos.

- **Histórico de Datos:** Mantener un registro histórico de los datos permite consultar y comparar el rendimiento de los autobuses a lo largo del tiempo, lo que es útil para la gestión de la flota y la planificación estratégica.

Seguridad de los Datos

Descripción: La transmisión de datos entre los autobuses y la central está protegida mediante encriptación y mecanismos de autenticación.

Justificación:

- **Privacidad e Integridad:** La encriptación asegura que los datos sean accesibles solo para usuarios autorizados y evita que sean alterados durante la transmisión. Esto protege la privacidad de la información y garantiza la integridad de los datos.
- **Cumplimiento Normativo:** La implementación de mecanismos de seguridad ayuda a cumplir con las regulaciones locales e internacionales sobre protección de datos y privacidad, evitando posibles sanciones y mejorando la confianza en el sistema.

Cumplimiento Normativo

Descripción: El sistema está diseñado para cumplir con las regulaciones locales e internacionales sobre la operación de vehículos de transporte, incluyendo restricciones de velocidad y otras normativas.

Justificación:

- **Evitación de Multas:** El cumplimiento de las normativas asegura que la empresa evite multas y sanciones asociadas con el incumplimiento de las regulaciones de transporte.
- **Seguridad en las Carreteras:** Al cumplir con las restricciones de velocidad y otras normativas.

Diseño de la arquitectura

Diagrama de Contexto General

El diagrama de contexto general proporciona una visión macro de los componentes y la interacción entre los distintos elementos del sistema de monitoreo y telemetría para "Buses Seguros S.A.". En este diagrama, se destacan los flujos de datos y las relaciones principales entre los sensores en los autobuses, los componentes del sistema en cada autobús, y los sistemas centrales en la empresa.

- **Sensores:** Recopilan datos sobre el estado del autobús (relación de medición).
- **Cluster Kubernetes:** Ejecuta pods para procesar los datos de los sensores (relación de procesamiento).
- **Colas de Mensajería:** Almacenan temporalmente los datos (relación de almacenamiento temporal).
- **API Gateway:** Recibe las solicitudes de datos desde los autobuses (relación de recepción).
- **Microservicios:** Procesan, almacenan y analizan los datos (relación de procesamiento y análisis).
- **Base de Datos:** Almacena los datos históricos y en tiempo real (relación de almacenamiento).
- **Sistema de Alertas:** Notifica a los operadores sobre eventos críticos (relación de notificación).

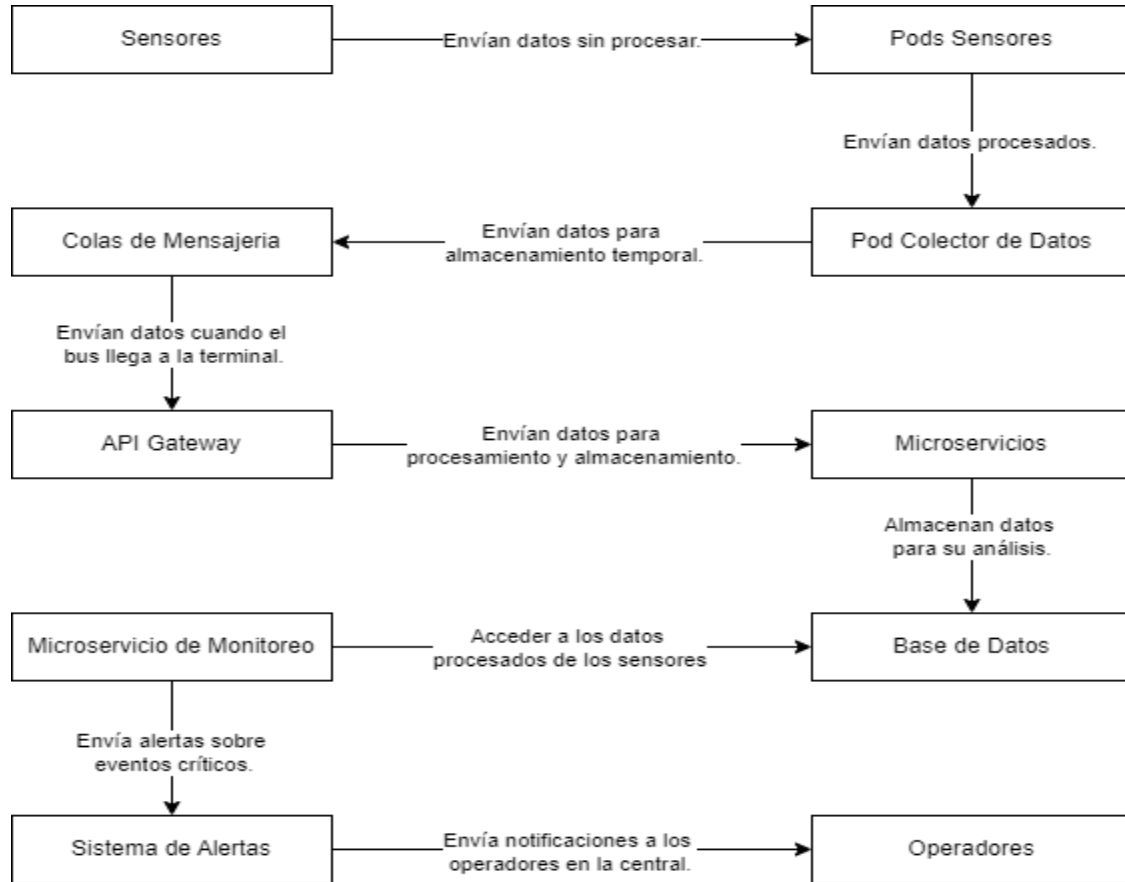


Diagrama de Componentes

El diagrama de componentes ilustra la estructura del sistema de monitoreo y telemetría para la empresa de transporte "Buses Seguros S.A.". Este sistema se despliega tanto en los autobuses como en la central de la empresa, permitiendo la captura, procesamiento, transmisión y análisis de datos en tiempo real.

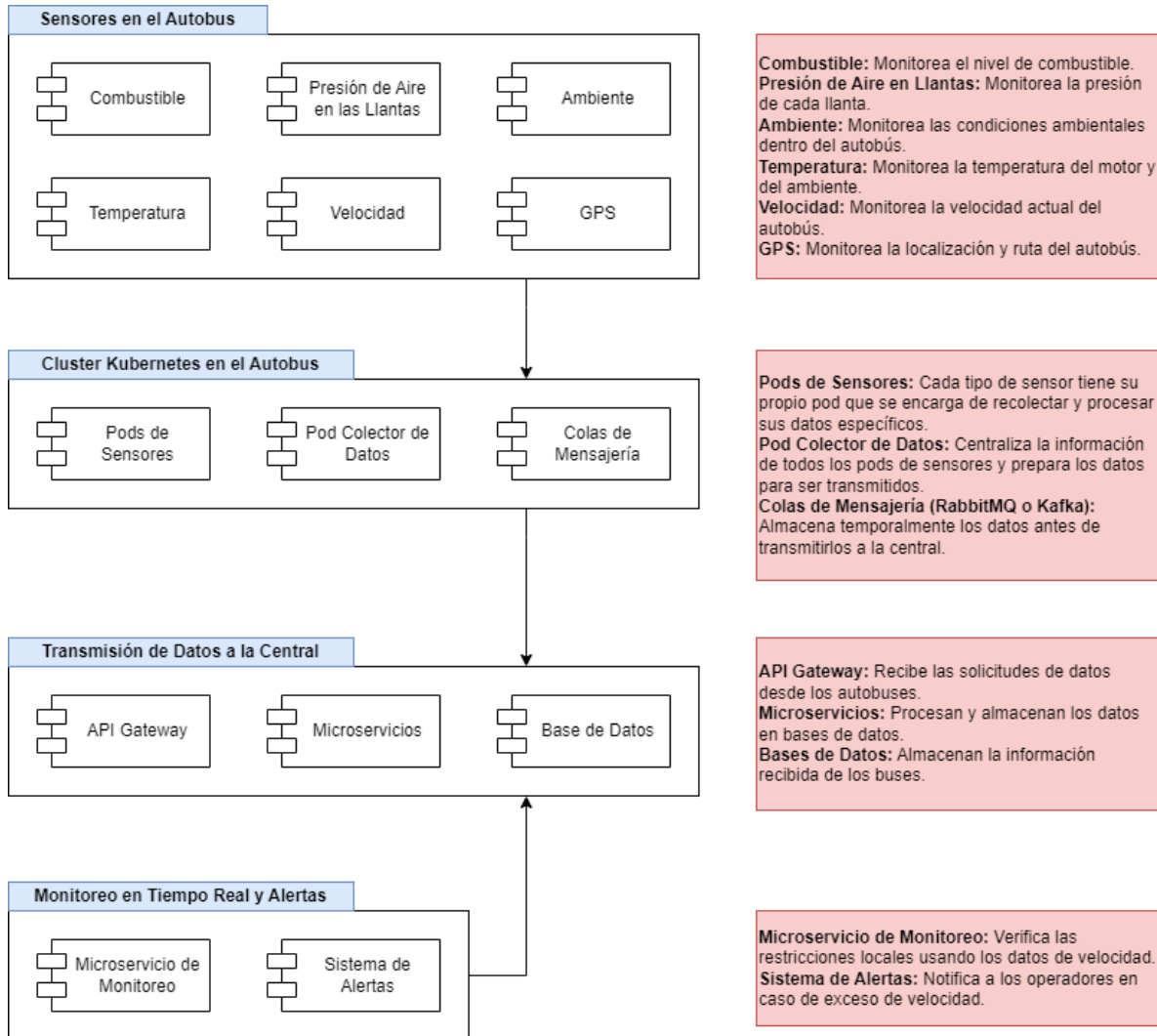


Diagrama de Despliegue

El diagrama de despliegue ilustra la arquitectura física del sistema de monitoreo y telemetría para la empresa de transporte "Buses Seguros S.A.". Los principales componentes y su interacción se dividen en dos entornos: el entorno del autobús y el entorno central de la empresa.



1. Terminal

- **API Gateway:** Recibe datos desde los autobuses.
- **Microservicios:** Procesan, almacenan y monitorean los datos.
- **Bases de Datos:** Almacenan la información procesada.

2. Autobús

- **Cluster Kubernetes:** Gestiona los pods en el autobús.
- **Pods (Sensor):** Recogen y procesan datos de diferentes sensores.
- **Pod Colector de Datos:** Centraliza y prepara los datos para la transmisión.
- **Cola de Mensaje (RabbitMQ/Kafka):** Asegura que los datos se almacenen temporalmente y se transmitan de manera confiable.
- **Sensores:** Dispositivos físicos que recogen datos sobre el autobús (combustible, presión, ambiente, temperatura, velocidad, GPS).

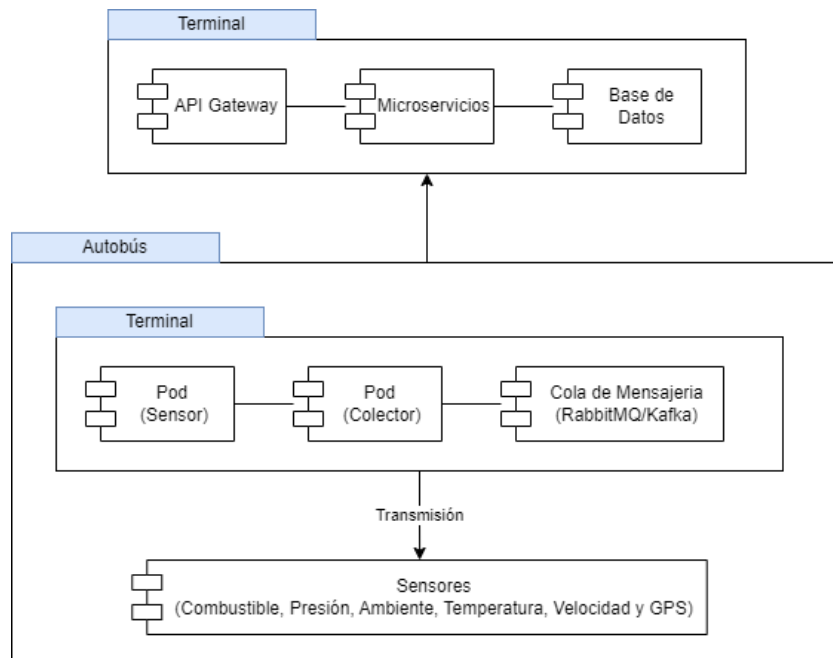


Diagrama de Secuencia

El diagrama de secuencia muestra las interacciones entre los componentes clave del sistema en una serie de pasos que reflejan el flujo de datos a través del sistema. Cada paso en el diagrama representa un componente o una acción específica en el proceso de monitoreo y telemetría:

1. Recolección y Procesamiento de Datos en el Autobús:

- **Sensores:** Los sensores instalados en el autobús recolectan datos sobre diversas métricas como combustible, presión de aire en las llantas, condiciones ambientales, temperatura, velocidad y ubicación GPS.
- **Pods de Sensores:** Los datos de los sensores se envían a los pods correspondientes dentro del cluster Kubernetes en el autobús. Cada pod está especializado en manejar datos de un tipo específico de sensor.
- **Pod Colector de Datos:** Este pod centraliza los datos provenientes de los diferentes pods de sensores y los prepara para su transmisión a través de las colas de mensajería.

2. Transmisión de Datos a la Central:

- **Colas de Mensajería:** Los datos preparados por el pod colector se envían a las colas de mensajería (RabbitMQ o Kafka), donde se almacenan temporalmente. Esto garantiza que los datos sean almacenados de manera confiable antes de su transmisión a la central.
- **API Gateway:** Cuando el autobús llega a una terminal, los datos almacenados en las colas de mensajería se transmiten a la central a través de un API Gateway. Este componente recibe los datos y los dirige a los microservicios adecuados para su procesamiento.

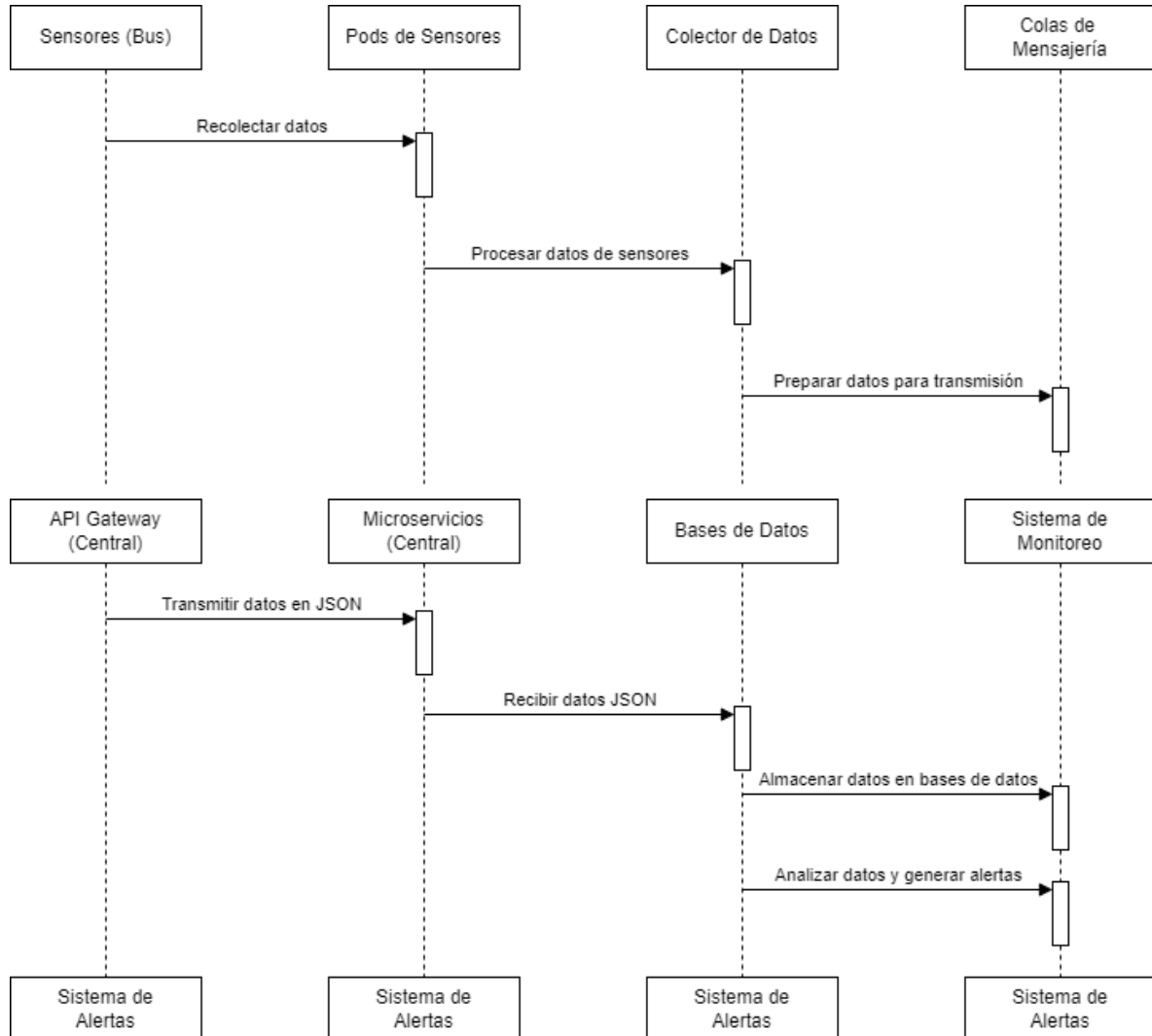
3. Procesamiento y Almacenamiento en la Central:

- **Microservicios:** En la central, los microservicios procesan y almacenan los datos recibidos en las bases de datos de la empresa. Este procesamiento incluye la preparación de los datos para su análisis y almacenamiento a largo plazo.
- **Bases de Datos:** Los datos se almacenan en bases de datos para su análisis posterior y la generación de informes.

4. Análisis y Generación de Alertas:

- **Sistema de Monitoreo y Alertas:** Los microservicios analizan los datos almacenados, especialmente la velocidad del autobús, para detectar cualquier violación de las restricciones locales. Si se detecta una infracción, el sistema de alertas genera notificaciones para los operadores en la central.

El diagrama de secuencia proporciona una visión clara de cómo los datos fluyen desde su origen en los sensores hasta su análisis y la generación de alertas, destacando la interdependencia entre los diferentes componentes del sistema y asegurando una operación eficiente y efectiva.



Diagramas C4



Diagrama de contexto

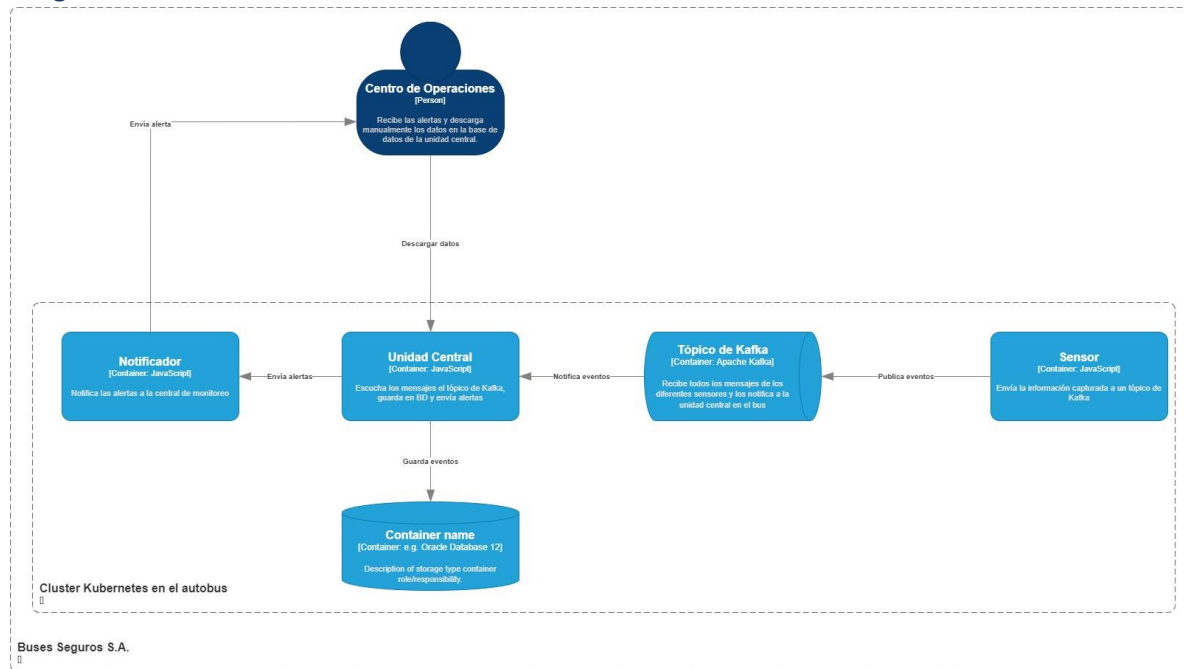




Diagrama de Contenedores

Sistema de Monitoreo y Telemetría - Diagrama de Contenedores

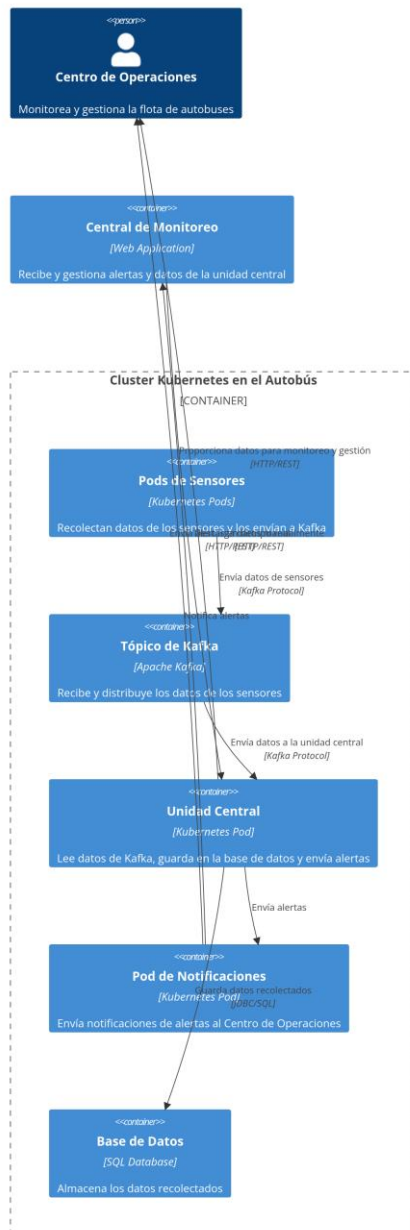


Diagrama de Componentes

Unidad Central - Diagrama de Componentes

