Documento de Especificaciones de Producto [DEP]

Proyecto: VisionGuard Solutions

Diciembre 2024

Instrucciones para el uso de este formato

Este formato es una plantilla tipo para documentos de requisitos de producto para su desarrollo.

Está basado y es conforme con el estándar IEEE Std 830-1998 y ha sido modificada para su suso en un ambiente de desarrollo mecatrónico simplificado.

El uso de este documento permite capturar la información relevante para desarrollar un producto o algunas de sus partes, sean electrónicas, mecánicas, de software o funcionales.

Las secciones que no se consideren aplicables al sistema descrito podrán de forma justificada indicarse como no aplicables (NA).

Notas:
Los textos en color azul son indicaciones que deben eliminarse y, en su caso, sustituirse por los contenidos descritos en cada apartado.

Los textos entre corchetes del tipo "[Inserte aquí el texto]" permiten la inclusión directa de texto con el color y estilo adecuado a la sección, al pulsar sobre ellos con el puntero del ratón.

Los títulos y subtítulos de cada apartado están definidos como estilos de MS Word, de forma que su numeración consecutiva se genera automáticamente según se trate de estilos "Titulo1, Titulo2 y Titulo3".

La sangría de los textos dentro de cada apartado se genera automáticamente al pulsar Intro al final de la línea de título. (Estilos Normal indentado1, Normal indentado 2 y Normal indentado 3).

El índice del documento es una tabla de contenido que MS Word actualiza tomando como criterio los títulos del documento.
Una vez terminada su redacción debe indicarse a Word que actualice todo su contenido para reflejar el contenido definitivo.

Ficha del documento

Fecha	Revisión	Autor	Verificado dep. calidad.
12/09/2024	1	Nicole Mendez	VisionGuard Solutions Katrina Arias

Documento validado por las partes en fecha: 7/12/2024

Por el cliente	Por la empresa suministradora
Fdo. D./ Dña [American inc.]	Fdo. D./Dña [VisionGuard Solutions]]

Contenido

FICHA DEL DOCUMENTO 3

CONTENIDO 4

1 INTRODUCCIÓN 6

1.1 Propósito 6

1.2 Alcance 6

1.3 Personal involucrado 7

1.4 Definiciones, acrónimos y abreviaturas 7

1.5 Referencias 8

1.6 Resumen ¡Error! Marcador no definido.

2 DESCRIPCIÓN GENERAL 8

2.1 Perspectiva del producto 8

2.2 Funcionalidad del producto 9

2.3 Características de los usuarios 9

2.4 Restricciones 9

2.5 Suposiciones y dependencias 11

2.6 Evolución previsible del sistema 13

3 REQUISITOS ESPECÍFICOS ¡ERROR! MARCADOR NO DEFINIDO.

3.1 Requisitos comunes de los interfaces 15
3.1.1 Interfaces de usuario 15
3.1.2 Interfaces de hardware 15
3.1.3 Interfaces de software 17
3.1.4 Interfaces de comunicación 18

3.2 Requisitos funcionales 20
3.2.1 Requisito funcional 1 **¡Error! Marcador no definido.**
3.2.2 Requisito funcional 2 **¡Error! Marcador no definido.**
3.2.3 Requisito funcional 3 **¡Error! Marcador no definido.**
3.2.4 Requisito funcional 4 **¡Error! Marcador no definido.**

3.3 Requisitos no funcionales 22
3.3.1 Requisitos de rendimiento 22
3.3.2 Seguridad 23
3.3.3 Fiabilidad 23
3.3.4 Disponibilidad 24
3.3.5 Mantenibilidad 24
3.3.6 Portabilidad 25

3.4 Otros requisitos ¡Error! Marcador no definido.

4 APÉNDICES ¡ERROR! MARCADOR NO DEFINIDO.

1 Introducción

VisionGuard Solutions es una empresa especializada en la instalación, mantenimiento y personalización de sistemas de videovigilancia para hogares, negocios y espacios industriales. Nos enfocamos en ofrecer soluciones de alta calidad que integran tecnología de punta para garantizar la seguridad y tranquilidad de nuestros clientes.

1.1 Propósito

El propósito de este documento es detallar el diseño, los servicios, y la implementación de una empresa dedicada a la instalación de sistemas de cámaras de seguridad. Incluye la descripción de los servicios ofrecidos, el mercado objetivo, las ventajas competitivas y el plan estratégico inicial para establecer la empresa en el mercado. Este documento servirá como una guía para tomar decisiones estratégicas, operativas y de marketing, además de definir los lineamientos para el lanzamiento y crecimiento del negocio.

Audiencia a la que estará dirigido:

- **Inversionistas y socios potenciales**
- **Directivos y equipo de liderazgo**
- **Consultores y asesores externos**
- **Clientes corporativos iniciales**

1.2 Alcance

El proyecto consiste en la creación y puesta en marcha de una empresa denominada VisionGuard Solutions, dedicada a ofrecer soluciones integrales en sistemas de videovigilancia. Los principales productos y servicios para desarrollar incluyen:

1. **Sistemas de cámaras de seguridad:**
    - **Cámaras IP de alta resolución.**
    - **Cámaras con visión nocturna, detección de movimiento y tecnología PTZ (Pan-Tilt-Zoom).**
    - **Cámaras con inteligencia artificial (reconocimiento facial y análisis de patrones).**
2. **Sistemas de monitoreo remoto:**
    - **Aplicaciones móviles y plataformas en la nube para supervisión en tiempo real. ○ Integración de alertas por correo electrónico o notificaciones en dispositivos inteligentes.**
3. **Servicios adicionales:**
    - **Instalación personalizada y diseño de sistemas adaptados a las necesidades del cliente.**
    - **Mantenimiento preventivo y correctivo de los equipos.**
    - **Capacitación para el uso y manejo de las plataformas de videovigilancia.**

Este proyecto se alinea con las definiciones y estándares establecidos en documentos de mayor nivel relacionados con la planificación estratégica del negocio, el análisis del mercado de sistemas de seguridad y las normas técnicas internacionales aplicables a sistemas de videovigilancia (por ejemplo, ISO/IEC 62676 para sistemas de CCTV). Además, se garantiza la consistencia con los siguientes parámetros clave:

- **Cumplimiento normativo: Respetar regulaciones locales e internacionales relacionadas con la privacidad, protección de datos y videovigilancia.**
- **Estandarización de procesos: Uso de manuales y lineamientos establecidos para la instalación, configuración y mantenimiento de equipos.**
- **Sostenibilidad financiera: Asegurar que las inversiones y operaciones estén alineadas con proyecciones de crecimiento definidas en los documentos estratégicos iniciales.**

1.3 Personal involucrado

Nombre	Katrina Arias
Rol	Dirección General
Categoría profesional	Técnico Superior
Responsabilidades	Definir la visión, misión y objetivos estratégicos de la empresa.
Información de contacto	829-000-000

Nombre	Thays Nivar
Rol	Seguridad Electrónica
Categoría profesional	Ingeniero
Responsabilidades	Diseñar sistemas de videovigilancia personalizados según las necesidades del cliente.
Información de contacto	829-000-000

Nombre	Nicole Mendez
Rol	Especialista en Soporte Técnico
Categoría profesional	Ingeniero
Responsabilidades	Brindar asistencia técnica remota y presencial a los clientes.
Información de contacto	829-000-000

1.4 Definiciones, acrónimos y abreviaturas

Definiciones

- **Cámara de seguridad IP:** Dispositivo de videovigilancia que utiliza una conexión a Internet para transmitir y almacenar datos.
- **Sistema de videovigilancia:** Conjunto de cámaras, dispositivos de grabación y software que permiten monitorear y registrar actividades en un área específica.
- **PTZ (Pan-Tilt-Zoom):** Tecnología que permite a una cámara moverse horizontalmente (pan), verticalmente (tilt) y hacer zoom, controlada manualmente o automáticamente.
- **Reconocimiento facial:** Tecnología basada en inteligencia artificial que identifica y verifica identidades mediante características faciales.
- **Mantenimiento preventivo:** Actividades realizadas regularmente para garantizar el correcto funcionamiento de los sistemas y prevenir fallas.
- **Monitoreo en tiempo real:** Supervisión de imágenes y datos capturados por cámaras al momento de su transmisión.

Acrónimos y Abreviaturas

- **AI (Artificial Intelligence):** Inteligencia Artificial, tecnología que simula procesos de pensamiento humano en máquinas.
- **CCTV (Closed-Circuit Television):** Televisión de circuito cerrado, sistema de cámaras utilizado para vigilancia.
- **DVR (Digital Video Recorder):** Grabador de video digital, dispositivo que almacena las imágenes captadas por las cámaras.
- **NVR (Network Video Recorder):** Grabador de video en red, utilizado para cámaras IP.
- **IP (Internet Protocol):** Protocolo de Internet, tecnología utilizada para conectar dispositivos a una red.
- **IoT (Internet of Things):** Internet de las cosas, concepto que conecta dispositivos físicos a Internet para recopilar y compartir datos.
- **HD (High Definition):** Alta definición, estándar de resolución de video que ofrece mayor calidad de imagen.

- **ROI (Return on Investment):** Retorno de inversión, indicador financiero que mide la rentabilidad de una inversión.
- **LAN (Local Area Network):** Red de área local, sistema que conecta dispositivos dentro de una ubicación específica.
- **UPS (Uninterruptible Power Supply):** Sistema de alimentación ininterrumpida, utilizado para garantizar energía continua a los equipos.

1.5 Referencias

Referencia	Titulo	Ruta	Fecha	Autor
ISO/IEC 62676	ISO/IEC 62676 - Sistema de Videovigilancia por CCTV	[Ruta]	2021	Organización Internacional de Normalización (ISO)
ISO/IEC 27001:2013	Norma ISO/IEC 27001 - Gestión de Seguridad de la Información		2013	ISO
Manual de configuración y especificaciones técnicas (serie específica según modelos adquiridos).	Manual Técnico de Cámaras Hikvision		2023	Hikvision Technologies
Edición 2022, compatible con NVR serie DHI.	Guía de Instalación y Configuración de NVR Dahua		2022	Dahua Technology
Publicada en el Diario Oficial de la Federación.	Ley General de Protección de Datos Personales en Posesión de los Particulares (México)		5 de julio de 2010	Instituto Nacional de Transparencia, Acceso a la Información y Protección de Datos Personales (INAI).

2 Descripción general

2.1 Perspectiva del producto

El sistema de cámaras de seguridad que proporciona VisionGuard es **un subsistema dentro de un sistema mayor de seguridad y gestión tecnológica integral**. Este sistema mayor incluye componentes como:

- **Control de accesos:** Lectores biométricos, tarjetas de proximidad, y otros dispositivos para restringir y monitorear entradas y salidas.
- **Sistemas de alarma:** Para detectar intrusiones, emergencias como incendios, o problemas en las instalaciones.
- **Seguridad perimetral:** Sensores de movimiento, cercas eléctricas, y otros dispositivos para proteger el perímetro.

- **Gestión centralizada:** Software y centros de monitoreo que integran los datos de todos los subsistemas para un control eficiente.

2.2 Funcionalidad del producto

El sistema de cámaras de seguridad desarrollado por VisionGuard debe realizar las siguientes funciones esenciales, organizadas de manera clara y comprensible para el cliente: **1. Monitoreo en Tiempo Real**

- Captura y transmisión de video en vivo desde todas las cámaras instaladas.
- Acceso remoto a las imágenes a través de aplicaciones móviles o software de escritorio.

2. Grabación y Almacenamiento

- Almacenamiento continuo o programado de las grabaciones en un NVR (Grabador de Video en Red) o en la nube.
- Gestión de archivos grabados para búsqueda y reproducción fácil por fecha, hora, o eventos específicos.

3. Detección de Movimiento y Alertas

- Activación automática de grabación y envío de alertas al detectar movimiento en áreas específicas.
- Integración con sistemas de alarma para emitir notificaciones por correo, SMS o aplicaciones.

4. Integración con Otros Subsistemas

- Conexión con control de accesos, alarmas, y sistemas perimetrales para una solución integral.
- Sincronización con sistemas de reconocimiento facial o detección de matrículas para mayor funcionalidad.

5. Gestión Centralizada

- Panel de control que permite administrar todas las cámaras y subsistemas conectados desde un único lugar.
- Configuración remota de parámetros, como resolución de video, zonas de detección, y horarios de grabación.

6. Mantenimiento Predictivo y Preventivo

- Detección de fallos en cámaras o componentes del sistema.
- Envío de alertas preventivas para asegurar la operatividad continua del sistema.

2.3 Características de los usuarios

Tipo de usuario	Gerentes y directores
Formación	Títulos universitarios en administración de empresas
Habilidades	toma de decisiones estratégicas
Actividades	Desarrollo de estrategias de seguridad

Tipo de usuario	Profesionales de Seguridad
Formación	ingeniería de seguridad
Habilidades	Evaluación de riesgos
Actividades	Análisis de riesgos y amenazas

Tipo de usuario	Técnicos de Soporte y Operadores
Formación	Formación técnica en electrónica
Habilidades	Instalación y mantenimiento de equipos
Actividades	Instalación de equipos de seguridad

2.4 Restricciones

1. Metodologías de Desarrollo

Limitaciones de tiempo y presupuesto: Dependiendo de los plazos de entrega y los recursos disponibles, se puede optar por metodologías ágiles (como Scrum o Kanban) para una entrega incremental y continua. Sin embargo, si el tiempo y los recursos son más limitados, se puede necesitar una metodología más estructurada, como el modelo en cascada, que requiere una planificación más rigurosa desde el principio.

Adecuación a la escalabilidad: Si el sistema está destinado a ser utilizado en diversos entornos (como pequeñas empresas o grandes corporaciones), se debe considerar la flexibilidad del enfoque metodológico para adaptarse a diferentes necesidades, lo que puede requerir el uso de prácticas ágiles para permitir la iteración y mejoras continuas.

2. Lenguajes de Programación

Compatibilidad con hardware y software: La elección del lenguaje de programación debe tener en cuenta las restricciones de hardware, como la necesidad de trabajar con sistemas embebidos o dispositivos de bajo consumo. En este caso, los lenguajes como C, C++ o Python son comúnmente utilizados en el desarrollo de sistemas de seguridad y protección, pero la compatibilidad con el hardware utilizado (como cámaras de vigilancia, sensores y dispositivos IoT) es crucial.

C y C++ son preferidos para sistemas embebidos y de bajo nivel debido a su eficiencia y control sobre los recursos del sistema.
Python es excelente para el desarrollo rápido de prototipos y análisis de datos (como en sistemas de monitoreo o aprendizaje automático).
JavaScript o TypeScript pueden ser esenciales si el sistema implica aplicaciones web o interfaces de usuario interactivas para monitoreo en tiempo real.
Eficiencia de los algoritmos: El sistema puede requerir la implementación de algoritmos complejos (como procesamiento de imágenes, reconocimiento facial, análisis de video o patrones en datos) lo que podría exigir lenguajes de alto rendimiento o el uso de bibliotecas especializadas. Esto también podría implicar la necesidad de optimizar los algoritmos para evitar una sobrecarga en el procesamiento o la memoria.

3. Normas y Regulaciones

Cumplimiento de normativas de seguridad: Dependiendo del tipo de datos que maneje el sistema (por ejemplo, datos personales, imágenes de videovigilancia, datos de acceso a instalaciones), es importante adherirse a normativas como el GDPR (Reglamento General de Protección de Datos) en la Unión Europea o la Ley de Protección de la Privacidad en diferentes países. Esto incluye:

Encriptación de datos: Asegurar que la transmisión y almacenamiento de datos sensoriales o personales sea segura y esté cifrada.
Gestión de accesos y autenticación: Implementación de control de acceso adecuado (como autenticación multifactor).
Conservación de registros: Asegurar que el sistema permita auditorías y tenga registros adecuados de accesos y acciones para cumplir con la legislación vigente. Normas de seguridad de sistemas: Si el sistema maneja infraestructura crítica (como centrales de energía, bancos, etc.), debe adherirse a normas internacionales de seguridad como ISO 27001 (gestión de seguridad de la información) y ISO 9001 (gestión de calidad). Estas normativas impactan tanto el desarrollo como la operación continua del sistema.

4. Restricciones de Hardware

Compatibilidad con dispositivos de monitoreo: Es crucial que el sistema sea compatible con una amplia gama de dispositivos de hardware (cámaras de seguridad, sensores de movimiento, controladores de acceso, etc.), muchos de los cuales pueden tener restricciones específicas de protocolo, comunicación o potencia.

Los dispositivos pueden requerir interfaces específicas (como ONVIF para cámaras de seguridad) o conectividad con protocolos de red como Wi-Fi, Ethernet, o LoRa (para IoT).

El hardware debe ser capaz de manejar una gran cantidad de datos en tiempo real sin comprometer el rendimiento, lo que podría requerir un procesamiento distribuido o el uso de tecnologías de edge computing.
Consumo de recursos: Los sistemas de monitoreo deben ser eficientes en cuanto al consumo de energía, especialmente si se utilizan en lugares remotos o fuera de la red eléctrica. Las restricciones de hardware, como la potencia limitada de las baterías en sistemas autónomos o el tamaño de los dispositivos, influirán en la elección de tecnologías y en la eficiencia del código.

Compatibilidad con dispositivos móviles: Si el sistema requiere aplicaciones móviles o interfaces de usuario para monitoreo remoto, la compatibilidad con sistemas operativos como iOS y Android también debe ser considerada. El desarrollo de aplicaciones móviles puede requerir el uso de tecnologías específicas, como React Native o Flutter.

5. Restricciones del Sistema Operativo

Compatibilidad con sistemas operativos embebidos: Si el sistema involucra dispositivos con sistemas embebidos (como cámaras o sensores IoT), puede ser necesario utilizar un sistema operativo específico como RTOS (Real-Time Operating System) para garantizar un procesamiento en tiempo real de las señales de los sensores y las respuestas del sistema.

Compatibilidad entre plataformas: Si el sistema debe funcionar en diversas plataformas (como servidores en la nube, PCs de escritorio o dispositivos móviles), se debe garantizar que el sistema sea compatible con diferentes sistemas operativos como Windows, Linux o macOS, así como versiones más recientes de sistemas operativos móviles.

Limitaciones de sistemas operativos móviles: En el caso de que el sistema de monitoreo y protección se controle a través de dispositivos móviles, las restricciones del sistema operativo (como la disponibilidad de APIs y permisos para acceder a recursos como cámaras o sensores) deben ser tomadas en cuenta.

6. Otros Factores

Escalabilidad y mantenibilidad: El diseño del sistema debe permitir su expansión sin comprometer el rendimiento ni la seguridad. Esto implica una arquitectura modular y un uso adecuado de patrones de diseño (como microservicios o arquitecturas basadas en contenedores) para garantizar que el sistema pueda crecer con facilidad y que se puedan hacer actualizaciones o modificaciones sin generar interrupciones en el servicio.

Interoperabilidad: Si el sistema se debe integrar con otros sistemas de seguridad existentes, debe diseñarse para ser interoperable, lo que puede incluir el soporte para diferentes APIs, protocolos de comunicación y estándares de la industria.

2.5 Suposiciones y dependencias

En el desarrollo de un sistema como el que ofrece **VisionGuard Solutions**, es fundamental identificar y definir claramente las suposiciones y dependencias que, si cambian, podrían afectar los requisitos y la viabilidad del proyecto. A continuación, se presentan algunos ejemplos clave:

Suposiciones:

1. **Disponibilidad del sistema operativo y hardware requerido:**

- **Suposición**: El sistema operativo (como Linux, Windows o RTOS) y los controladores necesarios estarán disponibles y serán compatibles con el hardware que se utilizará (cámaras, sensores, drones, etc.).
- **Impacto**: Si el sistema operativo o los controladores no están disponibles o no son compatibles, la arquitectura y el diseño del sistema deberán ajustarse para garantizar la compatibilidad, lo que podría llevar a retrasos en el desarrollo o la necesidad de elegir otro hardware o sistema operativo compatible.

2. **Disponibilidad de dispositivos de monitoreo y seguridad:**

   - **Suposición**: Los dispositivos de monitoreo (cámaras, sensores, alarmas) que se integrarán al sistema serán accesibles y estarán en el mercado dentro del plazo del proyecto. ○ **Impacto**: Si algún componente crítico (como cámaras de alta resolución o sensores de movimiento) no está disponible, se podría necesitar buscar alternativas que cumplan con requisitos similares o rediseñar el sistema para ajustarse a las nuevas tecnologías disponibles.

3. **Conectividad y recursos de red:**

   - **Suposición**: Se cuenta con una infraestructura de red estable y suficiente para soportar el volumen de datos generados por los sistemas de monitoreo (como cámaras en tiempo real o sensores IoT). ○ **Impacto**: Si la red disponible no es lo suficientemente robusta para manejar la cantidad de datos o la latencia es alta, esto podría afectar el rendimiento del sistema y la experiencia del usuario. En este caso, sería necesario optimizar los datos transmitidos o implementar tecnologías de edge computing para procesar los datos localmente.

4. **Normativas y regulaciones de seguridad vigentes:**

   - **Suposición**: Las leyes y regulaciones actuales de seguridad y protección de datos (como GDPR o normativas locales) no cambiarán significativamente durante el desarrollo del sistema.
   - **Impacto**: Si las regulaciones cambian durante el desarrollo o implementación, podría ser necesario modificar las características del sistema para garantizar su cumplimiento, lo que podría implicar ajustes en la gestión de datos, almacenamiento o en los controles de acceso.

5. **Capacidades de los usuarios finales:**

   - **Suposición**: Los usuarios finales (técnicos, operadores de seguridad, gerentes) tendrán el conocimiento básico y las habilidades necesarias para operar los sistemas de seguridad y monitoreo, y en caso de ser necesario, recibirán capacitación.
   - **Impacto**: Si los usuarios finales no tienen la formación esperada o no pueden adaptar el sistema a sus necesidades operativas, podrían surgir problemas de usabilidad o fallos operativos. En tal caso, el sistema podría requerir una interfaz de usuario más intuitiva o entrenamientos adicionales.

Dependencias:

1. **Dependencia de proveedores de hardware:**

- **Dependencia**: La disponibilidad y el rendimiento de los dispositivos de hardware (sensores, cámaras, drones, etc.) dependen de los proveedores y fabricantes seleccionados.
- **Impacto**: Si los proveedores de hardware no cumplen con los plazos de entrega o si los dispositivos no cumplen con las especificaciones de rendimiento, esto podría retrasar el proyecto y requerir la búsqueda de nuevos proveedores o la adaptación de soluciones.

2. **Dependencia de tecnologías de terceros:**

   - **Dependencia**: El sistema puede depender de tecnologías de terceros, como bibliotecas de software de procesamiento de imágenes, servicios de almacenamiento en la nube o plataformas de análisis de datos.
   - **Impacto**: Si las tecnologías de terceros experimentan cambios en sus políticas de precios, disponibilidad o actualización, esto podría requerir modificaciones en el diseño o en el software utilizado, afectando el costo o el tiempo de implementación.

3. **Dependencia de infraestructuras de red o almacenamiento en la nube:**

   - **Dependencia**: Si el sistema depende de la infraestructura de red para la transmisión de datos o de soluciones de almacenamiento en la nube para guardar la información de manera segura, su disponibilidad y rendimiento son factores clave para el éxito del proyecto.
   - **Impacto**: Problemas con la infraestructura de red (como caídas de red frecuentes) o cambios en las políticas de proveedores de almacenamiento en la nube pueden afectar el rendimiento y la fiabilidad del sistema, lo que puede requerir ajustes en la arquitectura de red o la infraestructura de almacenamiento.

4. **Dependencia de cumplimiento normativo por parte de las entidades regulatorias:**

   - **Dependencia**: El sistema debe estar alineado con las leyes de protección de datos y otras normativas del sector. ○ **Impacto**: Cualquier cambio en las normativas durante el desarrollo o implementación del sistema podría implicar la necesidad de rediseñar la arquitectura para cumplir con las nuevas regulaciones, lo que puede generar retrabajos y afectar el cronograma de implementación.

2.6 Evolución previsible del sistema

El sistema de VisionGuard Solutions está diseñado para ofrecer una solución robusta y flexible de seguridad y monitoreo, pero como cualquier sistema tecnológico, siempre existe la posibilidad de mejora y evolución. A continuación, se identifican algunas áreas clave donde podrían implementarse mejoras en el futuro, basadas en la evolución de las necesidades del mercado, avances tecnológicos y retroalimentación de los usuarios.

1. Mejora en el Rendimiento y Escalabilidad del Sistema

Optimización de la transmisión de datos: Si el sistema involucra la transmisión de grandes volúmenes de datos en tiempo real (como videos en alta definición de cámaras de seguridad), se podrían explorar mejoras en la compresión de datos o la implementación de algoritmos de procesamiento más eficientes para reducir la carga en la red y mejorar la latencia.

Edge Computing: Implementar tecnologías de computación en el borde (Edge Computing) para procesar los datos localmente, en lugar de depender exclusivamente de la infraestructura de la

nube. Esto permitirá una toma de decisiones más rápida y reducirá la dependencia de la conectividad a internet.

2. Integración con Nuevas Tecnologías

Inteligencia Artificial (IA) y Machine Learning (ML): Implementar capacidades avanzadas de IA y ML para mejorar el análisis de los datos capturados por los sistemas de seguridad. Por ejemplo:

- Reconocimiento facial para identificar intrusos en tiempo real.
- Análisis predictivo para anticipar posibles riesgos o amenazas basados en patrones históricos.
- Detección avanzada de anomalías en los flujos de datos para detectar intrusiones o comportamientos sospechosos de manera automática.
- Internet de las Cosas (IoT): Expandir la integración de dispositivos IoT para crear una red de sensores más amplia y diversa que permita la recopilación de datos más detallados de las instalaciones y áreas monitoreadas (temperatura, humedad, movimiento, etc.).

3. Mejora en la Interfaz de Usuario (UI) y Experiencia de Usuario (UX)

Desarrollo de interfaces más intuitivas: Mejorar las interfaces de usuario para facilitar su uso tanto en dispositivos móviles como en pantallas de escritorio. Esto podría incluir un diseño más amigable, accesible y optimizado para diversos tipos de usuarios (técnicos, gerentes, etc.).

Realidad aumentada (AR): Integrar realidad aumentada para ofrecer una visualización más interactiva de las cámaras de seguridad y otros dispositivos en tiempo real, permitiendo a los operadores identificar rápidamente áreas de riesgo o potenciales fallos en el sistema.

4. Mejora en la Seguridad y Protección de Datos

Encriptación avanzada y autenticación multifactor: Mejorar las prácticas de seguridad mediante la implementación de tecnologías avanzadas de encriptación para proteger los datos almacenados y transmitidos, así como mecanismos de autenticación multifactor para los usuarios del sistema.

Cumplimiento normativo global: A medida que las normativas de protección de datos y ciberseguridad evolucionan, el sistema podría integrar nuevas funcionalidades para cumplir con normativas adicionales, como GDPR, CCPA o nuevas leyes locales relacionadas con la protección de la privacidad.

5. Soporte para Nuevos Dispositivos y Plataformas

Compatibilidad con nuevos tipos de hardware: El sistema podría ampliarse para incluir soporte para nuevos dispositivos de monitoreo y seguridad, como drones de última generación, sensores biométricos o nuevas cámaras con tecnologías mejoradas de visión nocturna o resolución.

Compatibilidad con plataformas de nube adicionales: Si se utilizan servicios de almacenamiento o procesamiento en la nube, se podría expandir la compatibilidad con otros proveedores de servicios en la nube como Google Cloud, Microsoft Azure o IBM Cloud, para ofrecer más opciones a los clientes y mejorar la resiliencia del sistema.

6. Funcionalidades de Análisis y Reportes Avanzados

Generación automática de reportes inteligentes: Implementar sistemas de generación de reportes automáticos basados en los datos recopilados, utilizando inteligencia artificial para ofrecer análisis más precisos y recomendaciones para mejorar la seguridad en las instalaciones.

Dashboards personalizables: Mejorar los paneles de control con dashboards personalizables que permitan a los usuarios crear vistas según sus necesidades específicas de seguridad y monitoreo, facilitando el análisis en tiempo real de datos relevantes.

7. Mejoras en la Integración y Automatización de Flujos de Trabajo

Automatización de respuestas a incidentes: Implementar sistemas de respuesta automática ante situaciones de riesgo (por ejemplo, el envío de alertas o la activación de alarmas cuando se detectan eventos sospechosos), lo que mejorará la eficiencia y rapidez en la gestión de incidentes.

Integración con sistemas de terceros: Ampliar la capacidad de integración del sistema con otros sistemas de gestión empresarial (como ERP, CRM, etc.) o sistemas de gestión de infraestructura (como alarmas, controles de acceso, etc.), creando un flujo de trabajo más cohesivo y automatizado.

8. Expansión en Funcionalidades de Monitoreo Remoto

Monitoreo remoto avanzado: Mejorar la capacidad de monitoreo remoto mediante el uso de apps móviles o interfaces web más avanzadas, permitiendo a los usuarios acceder a los datos y controlar el sistema desde cualquier ubicación, a través de plataformas seguras.

9. Mejoras en la Gestión de Incidentes y Respuesta

Sistema de gestión de incidentes en tiempo real: Crear un módulo de gestión de incidentes que permita a los operadores registrar, clasificar y priorizar eventos de seguridad, con la capacidad de asignar tareas a los responsables y hacer un seguimiento en tiempo real.

10. Capacidades de Monitoreo Ambiental y de Salud

Monitoreo de condiciones ambientales: Implementar sensores adicionales para monitorear las condiciones ambientales en las instalaciones, como niveles de CO2, temperatura, humedad y otros factores críticos para la seguridad.

Integración de salud y seguridad laboral: Desarrollar módulos adicionales que puedan monitorear la salud y seguridad de los empleados en tiempo real, especialmente en ambientes industriales o de alto riesgo, alertando sobre posibles riesgos de accidentes laborales.

2.7 Requisitos comunes de los interfaces

2.7.1 Interfaces de usuario

El diseño de la interfaz de usuario (UI) de VisionGuard Solutions debe garantizar que los usuarios finales puedan interactuar con el sistema de manera intuitiva, eficiente y segura. A continuación se detallan los requisitos clave de la interfaz de usuario para el producto, con énfasis en la funcionalidad, la accesibilidad y el diseño visual.

2.7.2 Interfaces de hardware

Las interfaces entre el sistema de **VisionGuard Solutions** y sus componentes de hardware deben ser diseñadas de manera que permitan una comunicación eficiente y segura. A continuación, se especifican las características lógicas clave de estas interfaces, incluyendo

aspectos de configuración que permitirán un rendimiento óptimo y una gestión eficaz de los dispositivos de hardware.

1. Interfaz de Comunicación entre el Sistema y las Cámaras de Seguridad

- **Protocolo de Comunicación**:
  - Las cámaras de seguridad estarán conectadas al sistema utilizando **protocolos estándar** como **RTSP (Real-Time Streaming Protocol)** para transmisión de video en vivo o **ONVIF (Open Network Video Interface Forum)** para la interoperabilidad entre diferentes marcas y modelos de cámaras.
  - Se utilizará **HTTP/HTTPS** para la gestión remota de la cámara y la configuración de parámetros como resolución, enfoque y ángulo de visión.

- **Características de Configuración**:
  - **Dirección IP estática o DHCP**: La cámara debe configurarse con una **dirección IP estática** o mediante un servidor **DHCP** para garantizar que la cámara siempre sea accesible por el sistema. ○ **Configuración de resolución y calidad de video**: Los usuarios podrán configurar la resolución de la cámara (por ejemplo, 1080p, 4K) y la tasa de fotogramas (FPS), lo que afectará la calidad de la transmisión de video.
  - **Autenticación y seguridad**: Se implementará un sistema de autenticación por **usuario y contraseña** para acceder a las cámaras de seguridad, y se utilizarán métodos de cifrado (SSL/TLS) para proteger la transmisión de datos de video.

- **Detección de Cámara**:
  - El sistema deberá ser capaz de detectar automáticamente las cámaras conectadas a la red a través de un escaneo de dispositivos **UPnP (Universal Plug and Play)** o mediante una búsqueda manual basada en la dirección IP.

2. Interfaz entre el Sistema y Sensores de Movimiento

- **Protocolo de Comunicación**:
  - Los sensores de movimiento estarán conectados al sistema mediante **protocolos inalámbricos** como **Zigbee**, **Z-Wave** o **Wi-Fi**, dependiendo de la infraestructura y las necesidades de comunicación. ○ En caso de sensores cableados, se utilizarán **protocolos de comunicación como Modbus o RS485** para garantizar una comunicación estable y de largo alcance.

- **Características de Configuración**:
  - **Umbral de Sensibilidad**: Los sensores de movimiento podrán configurarse para detectar un rango específico de movimiento, ajustando su sensibilidad según las necesidades del cliente. Esto puede incluir configuraciones como distancia de detección, rango de ángulo y tipo de movimiento (por ejemplo, humano, vehículo).
  - **Notificaciones y alertas**: Los sensores podrán generar notificaciones de alerta al sistema cuando detecten un movimiento. El sistema debe estar configurado para recibir y procesar estas alertas en tiempo real, enviando mensajes a los operadores si es necesario.

- **Integración con cámaras**: La activación del sensor de movimiento debe estar vinculada con las cámaras de seguridad, permitiendo que estas cámaras se activen automáticamente o ajusten su ángulo de visión en función de la ubicación del sensor.

3. Interfaz entre el Sistema y Drones de Seguridad (si se incluye)

- **Protocolo de Comunicación**:
  - Los drones de seguridad se conectarán al sistema mediante **Wi-Fi**, **4G/5G** o **LTE** para asegurar una comunicación de alta velocidad y bajo retraso, permitiendo el control remoto y la transmisión en vivo de las imágenes y videos capturados por las cámaras del drone.
  - El control del drone también puede incluir una interfaz de **API** basada en **MQTT** o **REST** para integración con la plataforma de seguridad del sistema.

- **Características de Configuración**:
  - **Trayectoria y programación de vuelo**: Los drones podrán configurarse para realizar patrullajes automáticos de acuerdo con trayectorias predeterminadas. Los usuarios podrán establecer puntos de inicio y finalización de la ruta de vuelo, así como puntos de interés a vigilar. ○ **Alerta en caso de intrusión**: Los drones estarán configurados para identificar automáticamente situaciones inusuales y enviar alertas a la plataforma en caso de que detecten movimientos sospechosos fuera de las áreas previamente configuradas.
  - **Geolocalización**: El sistema permitirá la configuración de **límites geográficos virtuales** (geofencing) para que los drones no salgan de una zona específica, y se pueda recibir una notificación si el drone se sale de estos límites.

2.7.3 Interfaces de software

En el caso de VisionGuard Solutions, el sistema debe integrarse con varios productos de software, tanto internos como externos, para asegurar que las funcionalidades del sistema sean completas y eficientes. A continuación, se detallan las interfaces de software necesarias, especificando los productos involucrados, sus propósitos y las definiciones de los interfaces.

1. Integración con Software de Gestión de Base de Datos (DBMS)
Descripción del Producto Software Utilizado:

MySQL o PostgreSQL (base de datos relacional)
Estas bases de datos se utilizarán para almacenar de manera segura la información sobre cámaras de seguridad, sensores, usuarios, alertas y configuraciones.
El software de base de datos debe ser confiable, escalable y permitir consultas rápidas y eficientes. Propósito del Interfaz:

El objetivo principal de esta integración es permitir que el sistema pueda guardar, recuperar y actualizar información crítica de manera eficiente.
Las consultas y operaciones en la base de datos son necesarias para mostrar los estados actuales de los dispositivos, eventos, alarmas y datos históricos de las cámaras.
Definición del Interfaz:

Contenido: El sistema interactuará con la base de datos mediante consultas SQL para acceder a información como eventos históricos, estados de las cámaras, configuraciones de los sensores y registros de usuario.

Formato: El interfaz será de tipo API RESTful o ODBC para consultas directas. Las respuestas de las consultas serán en formato JSON para facilitar la integración y manipulación de los datos en el sistema.
Ejemplo de consulta: SELECT * FROM cameras WHERE status='active';
2. Integración con Plataformas de Almacenamiento en la Nube Descripción del Producto Software Utilizado:

Amazon Web Services (AWS) S3, Google Cloud Storage, o Microsoft Azure Storage.
Se utilizará para almacenar grabaciones de video, imágenes de cámaras de seguridad y otros datos pesados, asegurando la disponibilidad y redundancia de los datos a largo plazo.
Propósito del Interfaz:

El propósito es permitir el almacenamiento y la recuperación eficiente de los datos de video y las imágenes grabadas.
También se usará para el respaldo de los datos críticos del sistema y la integración con políticas de recuperación ante desastres. Definición del Interfaz:

Contenido: El sistema debe poder enviar y recibir archivos multimedia (videos, imágenes de cámaras, archivos de configuración) a la nube.
Formato: La transferencia de archivos se realizará utilizando las API de los servicios en la nube, como AWS SDK para Python, Google Cloud Storage API o Azure Blob Storage API.
Ejemplo de API para cargar un archivo:
Amazon S3: aws s3 cp video.mp4 s3://bucket-name/videos/
Google Cloud Storage: gsutil cp video.mp4 gs://bucket-name/videos/ Los datos estarán en formato de archivo de video estándar, como MP4, y las respuestas del servidor estarán en formato JSON.
3. Integración con Software de Autenticación y Gestión de Usuarios Descripción del Producto Software Utilizado:

OAuth 2.0 y LDAP (Lightweight Directory Access Protocol) o Active Directory. Estos sistemas permitirán gestionar el acceso de usuarios, realizar autenticación segura y autorizar diferentes roles dentro del sistema.
Propósito del Interfaz:

El propósito es permitir un sistema de autenticación unificado y seguro, con múltiples niveles de acceso, garantizando que solo usuarios autorizados puedan acceder a ciertas funcionalidades del sistema.
También se utilizará para integrar el sistema con los servicios existentes de gestión de identidades y roles dentro de la organización.
Definición del Interfaz:

Contenido: El sistema de seguridad se integrará con OAuth o LDAP para verificar la identidad del usuario a través de sus credenciales.
Formato: El interfaz utilizará tokens JWT (JSON Web Tokens) para autorizar las peticiones de los usuarios. Las respuestas serán de tipo JSON, y se manejarán mensajes de error como 401 Unauthorized si los usuarios no tienen permisos adecuados.
Ejemplo de autenticación mediante OAuth 2.0:
El sistema redirige a una página de autenticación de OAuth, y luego recibe un token de acceso en formato JWT para realizar operaciones posteriores.
Endpoint: POST /auth/token
Respuesta: { "access_token": "abc123" }

4. Integración con Software de Monitoreo de Red y Dispositivos (SNMP)
Descripción del Producto Software Utilizado:

Simple Network Management Protocol (SNMP).
Este protocolo será utilizado para supervisar el estado de los dispositivos conectados, como cámaras de seguridad, sensores y drones, y para gestionar sus configuraciones a través de la red.
Propósito del Interfaz:

El propósito es permitir al sistema monitorear el estado de los dispositivos de hardware (cámaras, sensores, drones) en tiempo real, capturando métricas como el uso de la CPU, la memoria y el estado de los dispositivos de red. La integración con SNMP proporcionará a los administradores del sistema visibilidad completa de la infraestructura y facilitará la resolución de problemas.
Definición del Interfaz:

Contenido: El sistema consultará dispositivos a través de SNMP para obtener métricas de rendimiento (por ejemplo, uso de CPU, memoria) y estados operativos (por ejemplo, si una cámara está activa o desconectada).
Formato: La comunicación será de tipo SNMP v2c o SNMP v3, con respuestas en formato OID (Object Identifier) que contendrán la información requerida.

2.7.4 Interfaces de comunicación

El sistema de VisionGuard Solutions debe ser capaz de comunicarse con otros sistemas de manera eficiente y segura, utilizando protocolos adecuados para garantizar una integración fluida, rápida y confiable. A continuación, se describen los requisitos específicos para las interfaces de comunicación con otros sistemas, así como los protocolos de comunicación que se utilizarán.

1. Requisitos Generales del Interfaz de Comunicación
Seguridad:

Todas las comunicaciones entre el sistema y los sistemas externos deben ser seguras, utilizando protocolos de cifrado como TLS (Transport Layer Security) o SSL (Secure Sockets Layer) para proteger los datos transmitidos contra interceptaciones o ataques.

Las autenticaciones basadas en OAuth 2.0 o JWT (JSON Web Tokens) deben ser empleadas para asegurar que solo los sistemas autorizados puedan comunicarse con el sistema.

Escalabilidad:

El sistema debe ser capaz de manejar múltiples conexiones simultáneas, especialmente en el caso de integración con sistemas de monitoreo o bases de datos grandes.

El protocolo debe ser eficiente en términos de consumo de ancho de banda y capacidad de procesamiento para permitir una expansión futura.

Fiabilidad y Redundancia:

Se deben implementar mecanismos de control de errores, como reintentos automáticos, verificación de integridad de los datos (mediante hashes o sumas de comprobación), y mensajes de error claros.

Las comunicaciones críticas, como las alertas de intrusión, deben ser entregadas sin fallos, por lo que se debe considerar la redundancia en los sistemas de red y servidores.

Interoperabilidad:

El sistema debe ser capaz de interoperar con una variedad de sistemas de terceros, sin importar el tipo de hardware o software que utilicen, utilizando protocolos abiertos y estándar.

2. Protocolos de Comunicación

A continuación, se especifican los principales protocolos que se utilizarán para la comunicación entre VisionGuard Solutions y otros sistemas.

2.1. Protocolo HTTP/HTTPS (Hypertext Transfer Protocol)

Uso: El protocolo HTTP o su versión segura HTTPS será utilizado para la comunicación básica entre el sistema de VisionGuard Solutions y las plataformas de control de usuario, paneles de administración y API de dispositivos (cámaras de seguridad, sensores, drones, etc.).

Descripción:

HTTPS se utilizará principalmente para asegurar la transmisión de datos entre el servidor y los clientes finales (usuarios, dispositivos), proporcionando un canal cifrado y protegiendo la información sensible.

El sistema estará expuesto a través de una API RESTful sobre HTTP/HTTPS, permitiendo que los sistemas externos interactúen con las funcionalidades del sistema.

Requisitos:

Las comunicaciones deben ser siempre cifradas utilizando TLS/SSL.

Se utilizarán métodos HTTP estándar como GET, POST, PUT y DELETE para la interacción con las APIs del sistema.

Los datos se intercambiarán principalmente en formato JSON o XML.

2.2. Protocolo MQTT (Message Queuing Telemetry Transport)

Uso: Se utilizará para la comunicación en tiempo real entre dispositivos de seguridad (como cámaras de seguridad y sensores) y el sistema central, especialmente en entornos donde se requiere una baja latencia y eficiencia en el consumo de ancho de banda.

Descripción:

MQTT es un protocolo de mensajería ligero y basado en publicación/suscripción, ideal para dispositivos de bajo consumo energético y sistemas IoT (Internet de las Cosas).

Permite la transmisión de datos como alertas de sensores, video en vivo, o datos de drones de forma eficiente.

Requisitos:

Los mensajes deben ser transmitidos a través de un Broker MQTT, y el sistema deberá tener capacidad de suscribirse a diferentes temas (por ejemplo, alertas de sensores, cámaras activadas).

Los datos de las alertas, estado de los dispositivos o imágenes deben ser enviados de forma comprimida para optimizar el uso del ancho de banda.

El sistema debe ser capaz de manejar la desconexión temporal y reconexión automática de los dispositivos.

2.8 Requisitos funcionales

2.8.1 Comprobación de validez de las Entradas

Antes de procesar cualquier información, el sistema debe asegurarse de que los datos recibidos sean válidos. Esta comprobación debe realizarse en cada paso del flujo de trabajo.

- **Entradas esperadas**: La información que debe ser procesada puede incluir datos de dispositivos (cámaras, sensores), comandos de los usuarios, configuraciones del sistema, entre otros.

- **Comprobación de validez**:

    - **Formato**: Verificación del formato de datos, como direcciones IP, identificadores de dispositivos, y datos numéricos. Por ejemplo, si un número de identificación es esperado como un valor numérico de 10 dígitos, el sistema debe verificar que se cumpla esta restricción.
    - **Rango de valores**: Asegurarse de que los valores de entrada estén dentro de los rangos permitidos, por ejemplo, valores de temperatura o humedad que no excedan los límites. ○ **Existencia**: Verificación de que los dispositivos o elementos referenciados existan en la base de datos antes de ejecutar cualquier operación.
    - **Autenticación**: Comprobación de las credenciales de acceso (usuario y contraseñas) y autorización basada en roles para garantizar que el usuario tenga permisos para realizar la acción solicitada.

2.8.2 Secuencia Exacta de Operaciones.

Una vez que las entradas han sido validadas, el software debe ejecutar una serie de operaciones en una secuencia lógica específica para cumplir con la tarea solicitada.

- **Operaciones generales**:

    - **Verificación de dispositivos activos**: Antes de procesar datos de cámaras o sensores, el sistema debe asegurarse de que dichos dispositivos estén activos y operativos. Si no lo están, se deberá generar una alerta y no realizar la operación solicitada.
    - **Procesamiento de datos**: Según el tipo de entrada (por ejemplo, movimiento detectado por una cámara o sensor de temperatura), el sistema debe generar las acciones apropiadas, como activar una alarma, almacenar el evento en la base de datos o enviar una notificación al usuario.
    - **Almacenamiento de datos**: La información relevante (eventos, alertas, configuraciones) debe ser almacenada de manera coherente en la base de datos según las reglas del sistema.

- **Secuencia**:

    1. Recepción de datos del dispositivo o del usuario.
    2. Validación de los datos.

3. Procesamiento según las reglas de negocio (por ejemplo, detección de intrusión o alerta de mal funcionamiento).
4. Almacenamiento de los resultados en la base de datos.
5. Generación de salidas o acciones a realizar (notificaciones, informes, comandos a dispositivos).

2.8.3 Respuesta a situaciones anormales.

El sistema debe manejar de manera adecuada cualquier situación anormal, como desbordamientos, fallos de comunicación o errores durante el procesamiento. Para ello, debe implementar una serie de mecanismos de recuperación y manejo de errores.

- **Desbordamientos**:
    - Si los datos recibidos exceden los límites establecidos, el sistema debe generar un mensaje de error e impedir que los datos sean procesados. Por ejemplo, si se reciben valores de temperatura fuera del rango de funcionamiento de los sensores, el sistema debe alertar al usuario de un posible error de medición.
- **Errores de comunicación**:
    - Si el sistema no puede establecer una conexión con un dispositivo o servidor de la base de datos, debe intentar reconectar durante un período determinado. Si la reconexión falla, se debe generar una alerta y registrar el error en los logs del sistema.
- **Recuperación de errores**:
    - En caso de fallos en el procesamiento (por ejemplo, pérdida de conexión con la base de datos), el sistema debe ser capaz de almacenar temporalmente los eventos o datos críticos y reintentar la operación de forma programada.
    - En caso de un error fatal o pérdida de datos, el sistema debe implementar mecanismos de **respaldo de datos** para asegurar la integridad de la información.

2.8.4 Parámetros

Los parámetros son valores clave que el sistema utiliza para personalizar y ejecutar operaciones de acuerdo con las necesidades del usuario o los requisitos del dispositivo.

- **Definición**:
    - **Parámetros de configuración**: Incluyen configuraciones como umbrales de sensores, duración de las grabaciones de video, tiempo de respuesta de alarmas, etc. ○ **Parámetros de dispositivos**: Definen la configuración de cada dispositivo, como la frecuencia de actualización de datos de sensores, la resolución de las cámaras, etc.
- **Proceso**:
    - Los parámetros son almacenados en la base de datos y pueden ser modificados por el usuario a través del panel de control. ○ El sistema debe

validar estos parámetros antes de aplicar cualquier cambio y asegurarse de que se ajusten a los límites establecidos por la configuración del dispositivo o los requisitos del sistema.

2.9 Requisitos no funcionales

2.9.1 Requisitos de rendimiento

El sistema de VisionGuard Solutions debe ser capaz de soportar una carga adecuada y ofrecer un rendimiento eficiente en condiciones normales y durante picos de uso. Los requisitos relacionados con la carga deben estar bien definidos para garantizar una experiencia de usuario fluida, sin demoras, y con una alta disponibilidad del sistema. A continuación, se detallan los requisitos específicos en cuanto a carga y rendimiento.

1. Número de Terminales (Dispositivos Conectados)

Requisito: El sistema debe ser capaz de manejar hasta 500 terminales (dispositivos como cámaras de seguridad, sensores y drones) conectados simultáneamente.

Descripción: Estos terminales pueden incluir cámaras IP, sensores de movimiento, sensores de temperatura, drones y otros dispositivos conectados al sistema. La comunicación con estos dispositivos debe ser continua, garantizando la sincronización y respuesta en tiempo real.

Medición: Se debe medir la capacidad del sistema para gestionar la comunicación con estos dispositivos sin pérdidas de datos ni demoras significativas.

2. Número de Usuarios Simultáneamente Conectados

Requisito: El sistema debe soportar al menos 200 usuarios simultáneamente conectados al panel de control web y aplicaciones móviles.

Descripción: Los usuarios pueden estar monitoreando el sistema, visualizando cámaras en vivo, recibiendo alertas y gestionando configuraciones. El sistema debe manejar la concurrencia de solicitudes de múltiples usuarios sin afectar el rendimiento.

Medición: La carga del sistema debe ser probada para asegurar que la plataforma web y la aplicación móvil pueden soportar este número de usuarios simultáneos sin experimentar lentitud o interrupciones en el servicio.

2.9.2 Seguridad

El sistema de VisionGuard Solutions debe ser capaz de soportar una carga adecuada y ofrecer un rendimiento eficiente en condiciones normales y durante picos de uso. Los requisitos relacionados con la carga deben estar bien definidos para garantizar una experiencia de usuario fluida, sin demoras, y con una alta disponibilidad del sistema. A continuación, se detallan los requisitos específicos en cuanto a carga y rendimiento.

1. Número de Terminales (Dispositivos Conectados)

Requisito: El sistema debe ser capaz de manejar hasta 500 terminales (dispositivos como cámaras de seguridad, sensores y drones) conectados simultáneamente.

Descripción: Estos terminales pueden incluir cámaras IP, sensores de movimiento, sensores de temperatura, drones y otros dispositivos conectados al sistema. La comunicación con estos dispositivos debe ser continua, garantizando la sincronización y respuesta en tiempo real.

Medición: Se debe medir la capacidad del sistema para gestionar la comunicación con estos dispositivos sin pérdidas de datos ni demoras significativas.

2. Número de Usuarios Simultáneamente Conectados

Requisito: El sistema debe soportar al menos 200 usuarios simultáneamente conectados al panel de control web y aplicaciones móviles.

Descripción: Los usuarios pueden estar monitoreando el sistema, visualizando cámaras en vivo, recibiendo alertas y gestionando configuraciones. El sistema debe manejar la concurrencia de solicitudes de múltiples usuarios sin afectar el rendimiento.

Medición: La carga del sistema debe ser probada para asegurar que la plataforma web y la aplicación móvil pueden soportar este número de usuarios simultáneos sin experimentar lentitud o interrupciones en el servicio.

2.9.3 Fiabilidad

La fiabilidad del sistema es un aspecto fundamental para garantizar que VisionGuard Solutions funcione de manera continua y sin fallos, minimizando el tiempo de inactividad y asegurando que los usuarios puedan acceder a las funcionalidades críticas de manera consistente. La fiabilidad se medirá en términos de tiempo entre fallos y el total de incidentes permisibles, y se establecerán umbrales específicos para asegurar que el sistema se mantenga operativo con el mínimo de interrupciones.

1. Tiempo Entre Fallos (MTBF - Mean Time Between Failures)

Requisito 1.1: MTBF (Tiempo Medio Entre Fallos) de al menos 1000 horas.

Descripción: El sistema debe estar diseñado de manera que el tiempo entre fallos, es decir, el promedio de horas de operación sin incidentes sea de al menos 1000 horas. Esto asegura que el sistema pueda funcionar durante largos períodos sin interrupciones importantes.

Medición: Se deben realizar pruebas de estrés y simulaciones de fallos para garantizar que el sistema alcance o supere este umbral de fiabilidad.

2. Tiempo Medio para la Recuperación (MTTR - Mean Time to Repair)

Requisito 2.1: MTTR (Tiempo Medio de Reparación) de menos de 2 horas.

Descripción: En caso de que ocurra un fallo, el tiempo necesario para restaurar el sistema y ponerlo nuevamente en funcionamiento no debe superar las 2 horas. Esto incluye la identificación del fallo, su resolución y la restauración del servicio.
Medición: El equipo de soporte debe estar preparado para reparar incidentes rápidamente, y se deben realizar pruebas periódicas para verificar la eficiencia de los tiempos de recuperación.

2.9.4 Disponibilidad

La disponibilidad del sistema es un factor crítico para VisionGuard Solutions, ya que el sistema debe estar operando de manera continua, minimizando el tiempo de inactividad y asegurando que los usuarios puedan acceder a las funcionalidades esenciales del sistema en todo momento. La disponibilidad se expresa generalmente en un porcentaje del tiempo total en el que el software debe estar operativo.

1. Disponibilidad General del Sistema

Requisito 1.1: Disponibilidad mínima del 99.9%.

Descripción: El sistema debe estar operativo el 99.9% del tiempo, lo que equivale a un máximo de 43 minutos de inactividad al mes o 8.76 horas de inactividad al año.

Medición: Se debe monitorizar continuamente el tiempo de inactividad para asegurar que no supere este umbral, incluyendo las interrupciones no planificadas y los mantenimientos programados.

2. Disponibilidad de los Componentes Críticos

Requisito 2.1: Disponibilidad mínima del 99.95% para los componentes críticos.

Descripción: Los componentes críticos del sistema, tales como la base de datos, el servidor de autenticación, y el servidor de control de dispositivos, deben tener una disponibilidad de al menos 99.95%. Esto implica un máximo de 22 minutos de inactividad al mes o 4.38 horas al año para estos componentes.

Medición: Estos componentes deben ser objeto de monitoreo intensivo y contar con redundancia (como servidores en clúster o balanceo de carga) para minimizar la posibilidad de fallos.

2.9.5 Mantenibilidad

El mantenimiento del sistema de VisionGuard Solutions es fundamental para asegurar su correcto funcionamiento a largo plazo, evitando fallos y garantizando la continuidad del servicio. A continuación, se detallan los tipos de mantenimiento necesarios, quién debe realizarlos, y cuándo deben llevarse a cabo.

1. Mantenimiento Correctivo

Descripción: Este tipo de mantenimiento se realiza para corregir errores o fallos en el sistema que afectan su funcionalidad o rendimiento. Es necesario cuando un incidente ocurre o se identifica un fallo en el sistema que requiere ser resuelto.

Responsable: Equipo de soporte técnico y desarrolladores del sistema.

Frecuencia: Este mantenimiento se realiza de manera ad hoc en respuesta a incidentes o fallos no previstos. Debe ejecutarse tan pronto como se identifique un problema.

Actividades Típicas:

- Diagnóstico y resolución de fallos del sistema (errores en la base de datos, fallos en la red, problemas con los sensores o drones).
- Corrección de errores de software que afectan la experiencia del usuario.
- Restablecimiento de la funcionalidad tras un fallo de hardware o software.

2. Mantenimiento Preventivo

Descripción: Este mantenimiento busca prevenir problemas y fallos en el futuro mediante la revisión periódica del sistema, actualización de componentes, y limpieza de recursos.

Responsable: Administradores de sistema y personal de infraestructura.

Frecuencia: Realización mensual o trimestral dependiendo de la criticidad de los componentes. Este mantenimiento debe estar programado para evitar interrupciones y minimizar el impacto en los usuarios.

Actividades Típicas:

Actualización de sistemas operativos, parches de seguridad y bibliotecas de software.
Revisión y optimización de la base de datos.
Verificación y actualización de los registros de seguridad y acceso.
Reemplazo de hardware obsoleto o con alta probabilidad de fallo.
Realización de pruebas de rendimiento y capacidad.

2.9.6 Portabilidad

La portabilidad del sistema es un requisito importante para VisionGuard Solutions, especialmente dado el entorno tecnológico en constante evolución y la necesidad de asegurar que el software pueda ejecutarse en diversas plataformas o entornos sin requerir modificaciones significativas. A continuación, se detallan los atributos clave que el software debe presentar para facilitar su traslado a otras plataformas.

1. Porcentaje de Componentes Dependientes del Servidor

Requisito 1.1: Los componentes dependientes del servidor deben ser lo más reducidos posible, con un objetivo de que menos del 30% del sistema dependa de un servidor específico o de infraestructura física. Esto permitirá mayor flexibilidad para mover el sistema entre diferentes servidores o entornos de nube.

Objetivo: Mantener la independencia del servidor, usando contenedores o microservicios para aislar los componentes del servidor.

Medición: Evaluar el porcentaje de dependencias directas del sistema con la infraestructura del servidor (bases de datos, servicios web, etc.).

2. Porcentaje de Código Dependiente del Servidor

Requisito 2.1: El código dependiente del servidor debe ser mínimo, con un objetivo de que menos del 20% del código fuente sea específico para un servidor o plataforma en particular. Este código debe ser fácilmente adaptable a otros entornos mediante el uso de abstractización o interfaces comunes.

Objetivo: Evitar dependencias específicas de hardware o infraestructura que dificulten el traslado a otro entorno.

Medición: Realizar un análisis de las bibliotecas y recursos utilizados por el código para verificar cuán portables son.

3. Uso de Lenguajes de Programación Portables

Requisito 3.1: El software debe estar desarrollado utilizando lenguajes de programación ampliamente portables, como Java, Python o C#. Estos lenguajes están diseñados para ejecutarse en múltiples plataformas y no dependen de un sistema operativo específico, lo que facilita su migración a otras plataformas.
Objetivo: Minimizar la dependencia de lenguajes o herramientas que solo se ejecutan en un sistema operativo o arquitectura de hardware específicos.

Medición: Comprobar que los principales componentes del software están escritos en lenguajes multiplataforma y no en lenguajes que dependen estrictamente de un sistema operativo o entorno específico.