Engineer

Portafolio

Servicios de ingeniería con ejemplos de trabajos realizados.

Engineer ApS se registró en 2025. La mayor parte del trabajo que se muestra aquí lo realizó antes nuestra fundadora dentro de equipos de clientes; una parte la ha entregado la propia empresa y otra se ha realizado después por cuenta ajena. Las mismas manos y los mismos estándares: indicamos qué corresponde a cada contexto en lugar de confundirlos.

Conozca el trabajo **en su contexto** — las decisiones, la entrega y los resultados comunicados.

Vea el catálogo completo

Encuentre experiencia que merezca aplicar

Utilice esta trayectoria para encontrar experiencia relevante para su proyecto.

Explore nuestros servicios Póngase en contacto →

Lea las decisiones detrás de la trayectoria

Las entradas describen trabajos entregados en distintos servicios. Lea una reseña para conocer la situación y las decisiones que los sustentan.

Casos prácticos Póngase en contacto →

Estos nombres **aparecen en la trayectoria que sigue** — nada exótico que su equipo deba heredar.

Python PostgreSQL TypeScript React GitHub Docker

Vea toda la pila tecnológica

Explorar por categoría

INGENIERÍA DE SOFTWARE

Ingeniería de backend
Ingeniería de frontend
Desarrollo integral
API e integración
Desarrollo web
Sistemas de diseño e interfaces
Diseño UX/UI

Pruebas y control de calidad · Optimización del rendimiento · Python · Internacionalización

DATOS Y BASES DE DATOS

Bases de datos · PostgreSQL · SQL · Ingeniería de datos · Flujos de datos (ETL/ELT) · Análisis de datos · Gobernanza de datos · Almacenes de datos · GIS / Información geoespacial

NUBE, INFRAESTRUCTURA Y DEVOPS

Nube · Contenedores (Docker/Kubernetes) · DevOps · Automatización y CI/CD · Infraestructura · Linux y servidores · Administración de sistemas · Redes y VPN · Monitorización y observabilidad · Fiabilidad y copias de seguridad · Seguridad · Arquitectura de plataformas · Migraciones y modernización

CONSULTORÍA Y LIDERAZGO

Dirección técnica · Dirección de equipos · Mentoría y acompañamiento · Metodologías ágiles y Scrum · Gestión de proyectos · Producto y requisitos · Partes interesadas e informes · Documentación · Arquitectura de soluciones · Marca y marketing

Todo lo que sigue **se ha entregado** — no son propuestas ni planes.

Cada entrada indica qué cambió el trabajo y enlaza los servicios que demuestra.

Qué ofrecemos Cuánto cuesta →

{ Ingeniería de software }

Arquitectura de plataformas y soluciones

Decidir cómo construir el sistema completo

La estructura del sistema se define primero sobre el papel para que las piezas encajen y los costes se conozcan antes de escribir código.

Concurrencia estricta Swift 6 sobre C

Adopté la concurrencia estricta completa de Swift 6 sin actores, conectando un bucle de eventos C bloqueante con el actor principal mediante un productor, un consumidor y un único orden, tras constatar que una tarea por evento pierde el orden del que depende la interfaz.

Un flujo de eventos limitado para una aplicación macOS

Limité los búferes de eventos y la generación de registros tras un incidente de macOS documentado con un crecimiento de 41 MB por segundo y 111 GB de páginas comprimidas en una máquina de 36 GB; una ejecución completa de pruebas documentada redujo las líneas de registro del núcleo de 610.996 a 1.411.

Cliente macOS nativo en Swift 6

Desarrollé un cliente de mensajería nativo para macOS en Swift 6 y SwiftUI — 11.141 líneas en 53 archivos — sobre la interfaz C de un núcleo Rust enlazado como biblioteca estática desde una revisión fijada.

Codificador QR propio en 522 líneas

Incorporé al repositorio un codificador QR — Reed-Solomon sobre GF(256), disposición fija de módulos y ocho patrones de máscara — en 522 líneas para evitar una tercera dependencia de ejecución, y lo verifiqué leyendo la matriz final con un decodificador desarrollado de forma independiente.

Generador de CV basado en base de datos

Desarrollé en Python un generador de CV, referencias, portafolios y cartas de presentación basado en base de datos — 41 módulos y 10.580 líneas — que genera seis formatos de salida desde una fuente SQLite de 23 tablas ensamblada mediante un flujo idempotente de 19 pasos.

Servicios públicos en un servidor pequeño

Alojé los servicios públicos de la empresa en un host de 512 MB y un solo núcleo — una forja git, un servidor web para siete dominios, Tor, dos servidores de protocolos alternativos y bloqueo de intrusiones — tomando los 464 MB de memoria utilizable como restricción de diseño; la automatización de copias de seguridad es una implementación aparte.

Infraestructura principal creada desde cero

Como una de las primeras incorporaciones, diseñé y construí desde cero toda la infraestructura principal y los procesos de soporte de una startup SaaS de energía verde, sentando las bases para un crecimiento rápido.

Infraestructura aún en servicio años después

Planifiqué e implementé nuevas funciones de infraestructura para sistemas internos y externos, con soluciones que siguieron funcionando años después con cambios mínimos.

Decisiones técnicas fundacionales de plataforma

Definí las decisiones fundacionales de la plataforma durante las semanas siguientes a la apertura del repositorio en noviembre de 2025 — la separación en capas, el acceso a datos centrado en la base de datos y el criterio de cero advertencias —, que seguían vigentes nueve meses después.

348 tablas en 34 esquemas Postgres

Modelé el dominio marítimo en 348 tablas normalizadas distribuidas en 34 esquemas PostgreSQL — profesionales, empresas, barcos, empleos, reseñas y el resto — con tablas de referencia SMALLINT y claves UUID v7.

Datos PostgreSQL centrados en funciones

Diseñé una capa de datos PostgreSQL centrada en funciones — 1.275 funciones almacenadas en 34 esquemas — para que cada lectura y escritura pase por una función a la que la base de datos pueda conceder acceso, en lugar de hacerlo por una tabla.

API migrada a Huma v2 con paridad del 100 %

Migré la API HTTP de Fiber a Huma v2 — 649 rutas y 760 operaciones — y alcancé y mantuve una correspondencia del 100 % entre las rutas que registra el servidor y la descripción OpenAPI que publica.

Cambio de contexto entre organizaciones

Diseñé un sistema de cambio de contexto entre organizaciones con localStorage en el cliente y una copia en cookies del servidor, que permite actuar en nombre de las organizaciones administradas y aplica autorizaciones de mínimo privilegio.

Paso directo de JSON de Postgres a la API

Diseñé una arquitectura de paso directo de JSON en la que las funciones PostgreSQL devuelven documentos JSON completos que la API en Go transmite sin cambios, eliminando la deserialización intermedia y desacoplando la interfaz de los cambios del esquema.

Una plataforma marítima con límites claros entre sistemas

Diseñé una plataforma marítima por capas que separaba una interfaz PWA en Next.js, una API en Go (Huma/Fiber) y una capa de funciones PostgreSQL, manteniendo toda la lógica de negocio en la base de datos.

Infraestructura simplificada fuera de AWS

Reorganicé la infraestructura de un cliente internacional, eliminando servicios sin uso, ajustando la capacidad y trasladando cargas fuera de AWS. La fundadora informa de una reducción del gasto en la nube por un factor de 10, sin pérdida de productividad.

Un producto cartográfico de principio a fin

Diseñé la arquitectura, desarrollé y mantuve infraestructura, procesamiento de datos y una aplicación de mapas durante 2 años, cubriendo tanto entregas de software como mantenimiento operativo.

Procesos renovados: 8.000 horas ahorradas

Reorganicé los procesos internos y ahorré 8.000 horas mediante mejoras en la arquitectura del software, los sistemas y la planificación.

Ingresos multiplicados por 10 con mapas GIS

Dirigí el desarrollo de una aplicación de mapas GIS, contribuí a multiplicar los ingresos por 10 y consolidé el producto como un activo de datos principal.

Un plan de infraestructura para 14 departamentos

Diseñé un marco integral de infraestructura para un instituto de investigación que abarcaba 14 departamentos, con módulos flexibles, flujos de datos unificados y estrategias de soporte estructuradas para facilitar su adopción a largo plazo.

{ Ingeniería de software }

Desarrollo de backend y API

El motor de su aplicación o sitio web

La parte que el usuario no ve: dónde se guardan los datos, qué reglas se aplican y cómo pueden comunicarse otros sistemas con ella.

La mitad sin utilizar del núcleo, implementada

Implementé la mitad del núcleo de mensajería que la aplicación no utilizaba — transferencia de copias de seguridad, mensajes temporales, edición y reenvío de mensajes, invitaciones verificadas, proxies y política de cifrado —, ejecutando cada prueba contra la biblioteca real, sin simulaciones.

Analizador que verifica cada llamada FFI

Escribí un analizador que lee la cabecera C real de 7.308 líneas y verifica cada punto de llamada, cada constante de enumeración y que toda clase propietaria de un puntero sea final, después de que una cabecera provisional escrita a mano permitiera compilar y enlazar llamadas a tres funciones eliminadas que causaban fallos al ejecutarse.

Todas las cadenas en un árbol de contenido

Trasladé todas las cadenas visibles para el usuario de Python a un árbol de 874 archivos de contenido en tres idiomas tras encontrar traducciones muertas cuya inutilidad no se veía y una comprobación que evaluaba silenciosamente solo un tercio de los logros.

Cada regla del analizador tratada como error

Activé como errores todas las reglas del analizador Python y resolví 1.815 hallazgos hasta llegar a cero, con un motivo escrito para cada una de las escasas excepciones y dos de ellas respaldadas por una comprobación, además de un comentario.

Generador: 981 pruebas y 92 % de cobertura

Mantuve el generador bajo 981 casos de prueba con un mínimo del 92 % de cobertura de ramas y las advertencias tratadas como errores, y verifiqué la idempotencia ejecutando dos veces todo el flujo de construcción desde un archivo vacío y exigiendo que la segunda pasada no cambiara nada.

Espacios de contratación y carrera marítima

Construí espacios de contratación y de carrera profesional para marinos; la instantánea de implementación registrada contiene 151 funciones almacenadas en 48 tablas para vacantes, candidaturas, certificados, progresión de rango y tiempo de embarque.

Capa social: publicaciones, noticias y grupos

Desarrollé la capa social de la plataforma — publicaciones, noticias, grupos, menciones, un grafo de seguidores y un resumen de ocasiones — sobre la misma capa de datos centrada en funciones que el resto del producto.

Monedero de fidelidad, ligas e insignias

Desarrollé el sistema de fidelidad y reputación — 67 funciones sobre un registro de 31 tablas, con ligas, insignias y una tienda de canje —, bloqueando la fila del saldo para cerrar la ventana de doble gasto.

Límites de tasa y protección contra abusos

Desarrollé controles contra abusos que deniegan el acceso ante fallos — 22 limitadores de tasa con Redis, Cloudflare Turnstile, idempotencia de solicitudes y bloqueo del origen — para rechazar bots y avalanchas de solicitudes sin confiar en quienes las envían.

Monitorización de errores y trazas propias

Desarrollé internamente la monitorización de errores y las trazas OpenTelemetry, en lugar de adquirirlas — saneamiento de cargas, detección de picos y regresiones, resolución de símbolos y una señal de actividad sintética —, con 11 vistas para operaciones.

Trabajo lento trasladado a una cola River

Saqué las tareas lentas del procesamiento de solicitudes y las trasladé a una cola River — 15 módulos de trabajadores, 8 tareas programadas y 20 tareas pg_cron — para responder a las solicitudes mientras el trabajo continúa en segundo plano.

Stripe y acceso por compras en aplicaciones

Desarrollé la capa de pagos y derechos de acceso — Stripe junto con las compras dentro de las aplicaciones de Apple y Google —, regulando el acceso al directorio, la búsqueda y la exportación mediante un modelo de 11 tablas comprobado en el servidor.

Pruebas para Go, SQL y navegador

Desarrollé una batería de pruebas automatizadas por capas — 981 pruebas Go, 543 especificaciones de interfaz y navegador y 494 pruebas de comportamiento SQL — con pruebas de mutación, pruebas basadas en propiedades y un control obligatorio de accesibilidad.

Aplicaciones Django para IPTV

Desarrollé y mantuve aplicaciones web Django para la plataforma IPTV, entregando nuevas funcionalidades y mejorando el rendimiento y la estabilidad.

Aplicación reforzada con CSP basada en nonces

Reforcé la aplicación con CSP basada en nonces, HSTS, cookies SameSite, roles de base de datos de mínimo privilegio y nuevas comprobaciones de derechos de acceso en el servidor.

Combinación recursiva guiada por un catálogo

Implementé una combinación recursiva de preferencias JSON almacenadas, guiada por un catálogo, para evitar fallos por claves ausentes a medida que evoluciona el esquema.

Correo con conmutación y registro de entregas

Desarrollé el correo electrónico como capacidad de la plataforma — tres proveedores con conmutación por error, webhooks de entrega, registros de envío y entrega, plantillas y campañas — con una comprobación de arranque que impide iniciar el sistema sin un proveedor.

Contrato de API generado desde la base de datos

Generé el contrato de la API desde la base de datos hacia las capas externas — OpenAPI, un cliente TypeScript tipado de 44.076 líneas, 61 manejadores simulados y los límites que aplica la interfaz — con una comprobación en cada paso que falla ante cualquier divergencia.

Desarrollo de informes de análisis de datos

Desarrollé un sistema de informes de análisis de datos e incrementé en un 400 % los ingresos trimestrales del software mediante informes PDF generados con Python.

Más de 30 proyectos GIS: desarrollo y soporte

Dirigí el desarrollo, despliegue y soporte de más de 30 proyectos GIS con PostgreSQL, Bash, Python, JavaScript, GDAL, ArcGIS, PostGIS y Mapbox.

Dirección del desarrollo integral de mapas GIS

Dirigí el desarrollo integral de mapas GIS, coordinando PostgreSQL, Mapbox, ReactJS y NodeJS para entregar una solución integrada.

8 proyectos Power BI con manuales

Entregué 8 proyectos de Power BI con manuales completos, integrando las herramientas de Microsoft Power BI con una API en NodeJS y Python FastAPI para el análisis y la visualización de datos.

Diseño y operación de 6 flujos ETL/ELT

Diseñé, implementé y administré 6 flujos ETL/ELT con Google BigQuery, MSSQL, PostgreSQL, scripts de Shell, PL/pgSQL y Transact-SQL, integrando los datos para su procesamiento eficiente mediante una API en Python.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Concurrencia estricta Swift 6 sobre C · Codificador QR propio en 522 líneas · Generador de CV basado en base de datos · Decisiones técnicas fundacionales de plataforma · Datos PostgreSQL centrados en funciones · API migrada a Huma v2 con paridad del 100 % · Cambio de contexto entre organizaciones · Paso directo de JSON de Postgres a la API · Una plataforma marítima con límites claros entre sistemas

{ Ingeniería de software }

Desarrollo de frontend

Las pantallas que utiliza la gente

Interfaces que cargan rápido, funcionan en el teléfono y se comportan igual en los navegadores que utilizan sus clientes.

Sistema de diseño macOS de 48 tokens

Desarrollé un sistema visual de 48 tokens sobre el material de vidrio propio de la plataforma y escribí el analizador que rechaza números sin significado declarado, tamaños de fuente fijados directamente y animaciones que ignoran la preferencia de movimiento reducido.

17 iconos con nombre para lectores de pantalla

Asigné nombres accesibles a todos los controles de la interfaz compuestos solo por un icono tras encontrar el botón de envío anunciado como «arrow up circle, button», y escribí el analizador que exige una etiqueta a no más de ocho líneas del icono.

Control de calidad de 10.242 líneas analizado

Apliqué un formateador y un analizador a 10.242 líneas de JavaScript del control de calidad tras constatar que eran el mayor bloque de código del repositorio y el único que ninguna herramienta revisaba; corregí 13 hallazgos sin suprimir ninguno.

Paquete CSS reducido de 87 KB a 31 KB

Reduje el paquete de estilos de 87 KB a 31 KB al retirar una fuente base64 y eliminé 756 KB en 22 archivos que se publicaban en cada despliegue sin que nada los referenciara.

Hoja de impresión de 637 líneas

Desarrollé una hoja de estilos de impresión de 637 líneas para cuatro comportamientos documentados de motores de renderizado, incluidos títulos integrados en el párrafo que se imprimían en blanco sobre blanco cuando el navegador prefería el tema oscuro.

Movimiento reducido que cumple la preferencia

Corregí el comportamiento de movimiento reducido tras encontrar once selectores que seguían animándose con esa preferencia activa, porque una regla universal que elimina las transiciones pierde en especificidad frente a cualquier regla con una clase.

Una capa completa ausente del tema claro

Detecté que al tema claro le faltaba una capa de superposición de página completa desde su creación al comprobar que ambos temas pintaran cada superficie compuesta con el mismo número de capas.

Sistema de color cerrado en 28 colores

Delimité el sistema de color a 28 colores documentados con un analizador que rechaza valores usados pero no documentados, documentados pero no usados, mal medidos, reescritos como literales o situados a una distancia perceptiva de 0,02 de otro existente.

Los fallos de accesibilidad que el verificador no detecta

Detecté y corregí 11 defectos de accesibilidad que el navegador presentaba como correctos: ocho enlaces de pie de página que seguían en el orden de tabulación pese a pointer-events, una línea temporal de desplazamiento que recortaba el colofón en tres páginas y un motor que ejecutaba 70 de sus 105 reglas.

Comprobaciones de accesibilidad en 32 páginas y ambos temas

Implementé comprobaciones de nivel AA de WCAG 2.2 en 32 páginas representativas en ambos temas de color, con un registro de conformidad, un criterio AAA comprobado y el cumplimiento parcial del criterio 1.4.13 declarado explícitamente.

27 tamaños de fuente por una escala de 6 pasos

Sustituí 27 tamaños tipográficos renderizados cuyos vecinos más próximos diferían un 0,6 % por una escala de tercera mayor de seis pasos, y escribí el analizador que rechaza un vigesimoctavo tamaño.

Una red de seguridad que ya no cuesta media hora

Reduje el tiempo del control de calidad del sitio basado en navegador de 1.636 segundos a 615 al ordenar las comprobaciones de mayor a menor duración y ejecutarlas en un grupo limitado a cuatro trabajadores, tras medir que el orden alfabético requería 320 segundos frente a 224.

Sitio Hugo multilingüe en 3 idiomas

Desarrollé un sitio estático multilingüe en Hugo con 108 archivos de plantillas, incluidos 53 parciales, que publica cada página en cuatro representaciones desde un árbol de contenido en tres idiomas, y también en siete subdominios especializados construidos desde el mismo árbol.

Panel interno de operaciones de la plataforma

Desarrollé el panel interno de operaciones — unas 40 rutas de administración y 128 componentes para reclamaciones de titularidad, moderación, indicadores de funcionalidades, caché y diagnóstico — para operar la plataforma sin acceder directamente a la base de datos.

Mantuvo un sitio Hugo y mejoró la UI del producto

Mantuve y mejoré el sitio web estático heredado en Hugo y participé en mejoras de UI/UX del producto principal de gestión de activos.

i18n en inglés y danés

Establecí un sistema de internacionalización en inglés y danés con 12.027 mensajes por configuración regional distribuidos en 458 archivos de espacios de nombres, con vocabulario controlado por análisis estático y un límite de 400 líneas por archivo.

Diseño integral de UI y UX del producto

Diseñé la UI y la UX del producto de principio a fin — cuadrículas del directorio, vistas alternativas de tarjetas y tablas, validadores de requisitos en tiempo real y navegación mediante rutas de ubicación.

Sistema de diseño 'Liquid Glass' de iOS 26

Diseñé un sistema visual basado en 'Liquid Glass' de iOS 26 y un inventario canónico de componentes, con reglas de análisis estático que impiden divergencias en la interfaz.

Soporte completo de Progressive Web App

Implementé soporte completo de Progressive Web App — instalable y utilizable sin conexión — con caché en tiempo de ejecución de Workbox mediante next-pwa.

Interfaz Next.js 16: SSR, SSG y CSR

Desarrollé la interfaz Next.js 16 con estrategias deliberadas de SSR, SSG y CSR, y una factoría reutilizable de páginas de servidor con precarga que elimina la cascada de solicitudes N+1 de cada página autenticada que pasa a utilizarla.

30 sitios web: de Photoshop a HTML

Diseñé 30 sitios web y convertí diseños de Photoshop a HTML para entregar soluciones originales y flexibles.

100 aplicaciones web con HTML/HTML5

Desarrollé 100 aplicaciones web con HTML/HTML5, CSS/SCSS, Django, WordPress y Joomla, para distintas necesidades de presencia en línea.

Participación en el diseño de interfaces

Participé en el diseño de interfaces de usuario para ofrecer interfaces intuitivas, visualmente cuidadas y fáciles de utilizar.

200 mejoras de UI/UX presentadas

Presenté 200 mejoras de UI/UX para la aplicación de mapas GIS y contribuí a multiplicar los ingresos por 10 mediante nuevas funcionalidades del software.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Un flujo de eventos limitado para una aplicación macOS · Cliente macOS nativo en Swift 6 · Decisiones técnicas fundacionales de plataforma · Cambio de contexto entre organizaciones · Una plataforma marítima con límites claros entre sistemas · Dirección del desarrollo integral de mapas GIS

{ Ingeniería de software }

Desarrollo integral de producto

Un producto que funciona, de principio a fin

Un equipo que lleva una idea del boceto a un producto al que sus clientes pueden acceder, sin pasarla entre tres proveedores.

Reclamaciones y decisiones de titularidad

Desarrollé el proceso completo de reclamación de titularidad de empresas: un usuario solicita una empresa, un administrador decide y la aprobación actualiza el grafo de autorizaciones que determina quién puede editar cada recurso.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Cliente macOS nativo en Swift 6 · Generador de CV basado en base de datos · Sitio Hugo multilingüe en 3 idiomas · Espacios de contratación y carrera marítima · Capa social: publicaciones, noticias y grupos · Panel interno de operaciones de la plataforma · Aplicaciones Django para IPTV · Decisiones técnicas fundacionales de plataforma · Interfaz Next.js 16: SSR, SSG y CSR · Una plataforma marítima con límites claros entre sistemas · Un producto cartográfico de principio a fin · 100 aplicaciones web con HTML/HTML5 · Más de 30 proyectos GIS: desarrollo y soporte · Dirección del desarrollo integral de mapas GIS · Ingresos multiplicados por 10 con mapas GIS

{ Ingeniería de software }

Desarrollo de sitios web y CMS

Un sitio web que puede actualizar su equipo

Un sitio rápido, fácil de encontrar y editable por su propio personal, sin una factura por cada cambio de texto.

Espejo onion de Tor con una dirección legible

Publiqué el sitio como espejo onion de Tor en una dirección legible que comienza por engineer, con la clave generada fuera del servidor para que nunca llegara a este repositorio ni a un portátil, y lo mantuve como la única de las cuatro puertas de entrada que nunca se contabiliza.

Identidad visual creada a partir de 2 dibujos

Desarrollé los conjuntos de iconos y favicons de la empresa a partir de dos dibujos originales, con todos los recursos derivados regenerados mediante scripts y una comprobación que falla si se confirma un archivo derivado anterior a su fuente.

106 de 106 comprobaciones del perfil PDF/UA-1

Registré la validación PDF/UA-1 en nueve documentos, con 106 de 106 comprobaciones superadas en el perfil de accesibilidad configurado, mientras la variante de archivo incumplía una de 146 reglas.

Sitemap corregido: fechas que eran iguales

Corregí un sitemap en el que 172 de 176 URL compartían una misma fecha de modificación, tomando la fecha del historial git tras constatar que la exportación reescribe todos los archivos en cada ejecución.

963 bloques de metadatos por un único grafo

Sustituí 963 bloques anónimos de datos estructurados que repetían la empresa 1.671 veces por un único grafo enlazado de 16 tipos con identificadores de origen estables.

Espejos del sitio en Gemini y Gopher

Construí salidas Gemini y Gopher a partir de la misma fuente del sitio web; el logro registra 777 documentos en cada formato en aquella etapa, con cero bytes de HTML modificados al añadir los formatos.

495 páginas generadas desde SQLite

Generé 495 páginas de logros en tres idiomas mediante una exportación SQLite de solo lectura, con las direcciones de página definidas como datos para que corregir una frase dejara de mover la página y romper su enlace.

Su código en su propio servidor

Desplegué la forja git propia de la empresa con Soft Serve, privada por defecto, sin panel web y con el puerto SSH limitado a loopback detrás de un servidor de salto, e hice de su página de entrada un artefacto de compilación del sitio principal en lugar de una copia mantenida a mano.

Desarrollo web combinado con SEO y contenidos

Impulsé los resultados web de los clientes combinando desarrollo y diseño de sitios a medida con SEO, estrategia de contenidos y redacción.

40 sitios web con Apache y Nginx

Administré 40 sitios web en servidores de alojamiento Ubuntu Linux con Apache y Nginx, manteniendo una disponibilidad y un rendimiento elevados.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Paquete CSS reducido de 87 KB a 31 KB · Hoja de impresión de 637 líneas ·

Movimiento reducido que cumple la preferencia ·

Los fallos de accesibilidad que el verificador no detecta ·

Comprobaciones de accesibilidad en 32 páginas y ambos temas · Sitio Hugo multilingüe en 3 idiomas ·

Mantuvo un sitio Hugo y mejoró la UI del producto · Aplicaciones Django para IPTV ·

Soporte completo de Progressive Web App · Interfaz Next.js 16: SSR, SSG y CSR ·

30 sitios web: de Photoshop a HTML · 100 aplicaciones web con HTML/HTML5

{ Ingeniería de software }

Preparación para agentes de IA

Aparecer cuando sus clientes preguntan a una IA

Un sitio o una aplicación que los asistentes de IA de sus clientes puedan encontrar, leer y utilizar, con comprobaciones en cada versión en lugar de una promesa inicial.

Desarrollo organizado en 51 iniciativas escritas

Organicé el desarrollo del sitio en 51 iniciativas escritas repartidas en 54 documentos de planificación, cada uno con tareas pendientes, mediciones y decisiones descartadas, incluidas seis de ocho conclusiones de preparación para agentes rechazadas con su motivo registrado.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Sitemap corregido: fechas que eran iguales · 963 bloques de metadatos por un único grafo · Comprobaciones de accesibilidad en 32 páginas y ambos temas · Sitio Hugo multilingüe en 3 idiomas

{ Ingeniería de software }

Herramientas para agentes de IA y servidores MCP

Permitir que la IA utilice el software que ya tiene

Su software actual disponible como herramientas que un agente de IA puede invocar, sin reescribir lo que ya funciona.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Panel interno de operaciones de la plataforma · Trabajo lento trasladado a una cola River · Contrato de API generado desde la base de datos · API migrada a Huma v2 con paridad del 100 % · Paso directo de JSON de Postgres a la API

{ Ingeniería de software }

Soluciones GIS y geoespaciales

Mapas que responden preguntas

Todo lo que ocurre en un lugar — barcos, vehículos, parcelas, sensores — representado en un mapa que se puede consultar, medir y compartir.

Entrega de 30 proyectos GIS satisfactorios

Gestioné la ejecución de 30 proyectos GIS satisfactorios, dirigiendo la entrega de soluciones innovadoras en este ámbito.

Optimización de previsiones e inversiones

Optimicé las estrategias de previsión e inversión de 11 operadores de redes eléctricas, mejorando la eficiencia operativa mediante soluciones GIS basadas en datos.

Entrega automática de 20 flujos de datos GIS

Automaticé la entrega de 20 flujos de datos GIS y procesos ETL de datos de aplicaciones, simplificando la automatización de la infraestructura y la elaboración de informes.

Despliegue GIS SaaS con GitHub Actions

Automaticé el despliegue de una aplicación GIS SaaS, el procesamiento de datos y el sistema de informes con GitHub Actions CI/CD, Python, Bash y SQL.

100 almacenes PostgreSQL, MS SQL y BigQuery

Diseñé la arquitectura, creé y administré 100 bases de datos de almacenes de datos en PostgreSQL, MS SQL y Google BigQuery, principalmente con datos GIS y series temporales, y optimicé su rendimiento y escalabilidad.

500 informes de datos eléctricos y GIS

Publiqué 500 informes de análisis de datos eléctricos y GIS, basados en investigación y resolución de problemas para ofrecer información precisa sobre grandes volúmenes de datos geográficos y series temporales.

1.000 incidencias de datos geográficos resueltas

Resolví 1.000 incidencias en datos geográficos y series temporales con GDAL, ArcGIS, PostGIS, Mapbox, QGIS, SQL (PL/pgSQL, Transact-SQL) y Bash, para mantener la calidad del procesamiento de grandes volúmenes de datos.

Mapas GIS 10 veces más rápidos con PostgreSQL

Multipliqué por 10 el rendimiento de una aplicación de mapas geográficos mediante una migración estratégica de MSSQL a PostgreSQL, incluida la transición de los datos, y optimicé el procesamiento y la seguridad de los datos.

Flujo de datos geográficos 50 veces más rápido

Registré una mejora de 50 veces en el flujo de procesamiento de datos geográficos en el trabajo de la fundadora para Utiligize mediante cambios en SQL y en los modelos de datos en PostgreSQL, MS SQL y Google Cloud BigQuery.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Un producto cartográfico de principio a fin · Más de 30 proyectos GIS: desarrollo y soporte · Dirección del desarrollo integral de mapas GIS · Ingresos multiplicados por 10 con mapas GIS

{ Ingeniería de software }

Diseño UI/UX y sistemas de diseño

Que resulte evidente cómo se utiliza

Distribución, textos y recorridos organizados para que las personas terminen lo que venían a hacer en vez de llamar a soporte.

Errores que dejan de culpar al usuario

Reescribí los mensajes de error de la aplicación conforme a un criterio de tono documentado después de que un inicio de sesión rechazado culpara al usuario de escribir mal la contraseña cuando el proveedor exigía una específica para aplicaciones, y lo cubrí con una prueba que identifica al proveedor.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Identidad visual creada a partir de 2 dibujos · Sistema de diseño macOS de 48 tokens · 17 iconos con nombre para lectores de pantalla · Cliente macOS nativo en Swift 6 · 106 de 106 comprobaciones del perfil PDF/UA-1 · Hoja de impresión de 637 líneas · Movimiento reducido que cumple la preferencia · Una capa completa ausente del tema claro · Sistema de color cerrado en 28 colores · Los fallos de accesibilidad que el verificador no detecta · Comprobaciones de accesibilidad en 32 páginas y ambos temas · 27 tamaños de fuente por una escala de 6 pasos · Desarrollo web combinado con SEO y contenidos · Mantuvo un sitio Hugo y mejoró la UI del producto · Diseño integral de UI y UX del producto · Sistema de diseño 'Liquid Glass' de iOS 26 · 30 sitios web: de Photoshop a HTML · Participación en el diseño de interfaces · 200 mejoras de UI/UX presentadas

{ Ingeniería de software }

Internacionalización y localización

Funcionar en más de un idioma

Software que se lee con naturalidad en cada mercado donde vende, incluidas las fechas, las monedas y la forma de ordenar los nombres.

Comprobación de cifras entre idiomas

Desarrollé una comprobación de contenido entre idiomas que falla si una traducción omite una cifra del inglés o usa una notación ajena al idioma; detectó dos descripciones danesas sin una métrica y dieciséis pasajes ucranianos con comillas del estilo inglés.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Todas las cadenas en un árbol de contenido · Generador de CV basado en base de datos · Espejos del sitio en Gemini y Gopher · 495 páginas generadas desde SQLite · Comprobaciones de accesibilidad en 32 páginas y ambos temas · Sitio Hugo multilingüe en 3 idiomas · i18n en inglés y danés

{ Ingeniería de software }

Desarrollo de agentes de IA a medida

Un asistente de IA que conoce su actividad

Un agente conectado a los sistemas que utiliza, probado con casos definidos por sus especialistas, supervisado por su personal y sin depender de un único modelo.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Seis recolectores declaran lo que omitieron · Pruebas para los propios verificadores · Control de commits con 22 analizadores · Monitorización de errores y trazas propias · Motor de 268 controles para el repositorio

{ Datos y bases de datos }

Diseño y modelado de bases de datos

Un lugar adecuado para guardar sus datos

Una estructura que refleja cómo funciona el negocio para poder responder también a las preguntas que surjan más adelante.

Analítica propia dentro de PostgreSQL

Desarrollé analítica de producto propia en PostgreSQL — 47 funciones sobre tablas de eventos particionadas que se depuran automáticamente —, seudonimizada con una sal rotatoria y supeditada al consentimiento del visitante.

Claves ordenadas temporalmente para un modelo de datos en crecimiento

Adopté identificadores UUID v7 ordenados temporalmente (PostgreSQL 18) como claves de las entidades para reducir la fragmentación de los índices B-tree y acelerar las consultas.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Generador de CV basado en base de datos · Espacios de contratación y carrera marítima · Monedero de fidelidad, ligas e insignias · Trabajo lento trasladado a una cola River · Stripe y acceso por compras en aplicaciones · 348 tablas en 34 esquemas Postgres · Combinación recursiva guiada por un catálogo · Datos PostgreSQL centrados en funciones · Paso directo de JSON de Postgres a la API · Una plataforma marítima con límites claros entre sistemas · 100 almacenes PostgreSQL, MS SQL y BigQuery · Diseño y operación de 6 flujos ETL/ELT

{ Datos y bases de datos }

Administración de bases de datos (DBA)

Cuidar de la base de datos

Una persona responsable del sistema en el que se apoya toda la empresa: que funcione, tenga copias de seguridad y no se deteriore sin que nadie lo advierta.

1.022 migraciones con control de coherencia

Mantuve la coherencia del esquema durante 1.022 migraciones con un control CI que crea la base de datos por dos vías — una instalación nueva y una instalación seguida de todas las migraciones — y falla si los resultados difieren.

Copias y recuperación ante desastres como código

Configuré la retención de las copias de seguridad de la base de datos como infraestructura como código, audité la capacidad de recuperación y documenté el procedimiento de restauración, dejando explícitas las carencias pendientes para evitar descubrirlas durante un incidente.

600 flujos ETL/ELT en PL/pgSQL

Desarrollé 600 flujos ETL/ELT basados en PL/pgSQL para simplificar procesos complejos de tratamiento de datos en varios entornos PostgreSQL de desarrollo y producción.

Datos automatizados con Python y PL/pgSQL

Automaticé tareas de procesamiento de datos con scripts de Shell, PL/pgSQL, Python y Transact-SQL, mejorando la productividad y la eficiencia.

100 copias críticas automatizadas con Barman

Automaticé 100 copias de seguridad de datos críticos con Barman, Google Cloud, Bash y Python, preservando la integridad de los datos entre bases de datos.

PostgreSQL 10 veces más rápido

Multipliqué por 10 el rendimiento de PostgreSQL mediante índices, particionamiento y optimización de consultas, mejorando la eficiencia de la base de datos para datos de usuarios, organizaciones cliente, información geoespacial y series temporales eléctricas.

1.000 reglas RBAC para servidores Linux

Reforcé la seguridad de los datos mediante 1.000 reglas RBAC para desarrolladores, instancias de aplicaciones, PostgreSQL, MS SQL y otros servidores Linux, evitando accesos no autorizados y documentando la configuración mediante automatización con Ansible.

10 servidores PostgreSQL y MS SQL en Ubuntu

Diseñé, desplegué y mantuve 10 servidores PostgreSQL y MS SQL en VPS con Ubuntu Linux, para mantener un rendimiento y una fiabilidad adecuados.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

100 almacenes PostgreSQL, MS SQL y BigQuery

{ Datos y bases de datos }

Optimización del rendimiento de bases de datos

Acelerar las partes que van lentas

Detectar las consultas y los índices que faltan en un sistema cada año más pesado y corregir lo que realmente importa.

30 días de datos eléctricos en 180 ms

Desarrollé variantes de agregación horaria del consumo eléctrico en Python, SQL y Bash + Jq, con una prueba en Python que procesó un conjunto JSONL de 30 días en 180 ms.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Claves ordenadas temporalmente para un modelo de datos en crecimiento ·

PostgreSQL 10 veces más rápido · 100 almacenes PostgreSQL, MS SQL y BigQuery ·

Mapas GIS 10 veces más rápidos con PostgreSQL · Flujo de datos geográficos 50 veces más rápido

{ Datos y bases de datos }

Migración y modernización de bases de datos

Dejar atrás el sistema antiguo

Trasladar los datos de un sistema envejecido a otro con soporte, manteniendo el negocio en funcionamiento durante el proceso.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Todas las cadenas en un árbol de contenido · 1.022 migraciones con control de coherencia ·

Mapas GIS 10 veces más rápidos con PostgreSQL

{ Datos y bases de datos }

Desarrollo de flujos de datos (ETL/ELT)

Que sus datos vayan de A a B por sí solos

Las cifras llegan donde hacen falta según un calendario, sin que nadie tenga que copiar una hoja de cálculo al terminar el mes.

4.000 horas ahorradas con CI/CD GitHub y GitLab

Simplifiqué los procesos de análisis de datos y desarrollo de software y ahorré 4.000 horas mediante prácticas CI/CD con GitHub, GitLab, Bash y Python.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

495 páginas generadas desde SQLite · Analítica propia dentro de PostgreSQL ·

Un producto cartográfico de principio a fin · 600 flujos ETL/ELT en PL/pgSQL ·

Datos automatizados con Python y PL/pgSQL · Más de 30 proyectos GIS: desarrollo y soporte ·

Entrega automática de 20 flujos de datos GIS · Despliegue GIS SaaS con GitHub Actions ·

Diseño y operación de 6 flujos ETL/ELT · 1.000 incidencias de datos geográficos resueltas · Flujo de datos geográficos 50 veces más rápido · 30 días de datos eléctricos en 180 ms · Un plan de infraestructura para 14 departamentos

{ Datos y bases de datos }

Diseño de almacenes de datos

Un lugar donde todas las cifras coinciden

Cifras de ventas, operaciones y finanzas reunidas para que dos departamentos no lleven dos respuestas distintas a la misma reunión.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

100 almacenes PostgreSQL, MS SQL y BigQuery · Diseño y operación de 6 flujos ETL/ELT · Flujo de datos geográficos 50 veces más rápido

{ Datos y bases de datos }

Análisis de datos y paneles de BI

Ver qué está ocurriendo

Paneles e informes con las pocas cifras de las que realmente depende una decisión, actualizados sin tener que pedirlo.

Sistema de gestión del tiempo e informes

Diseñé un sistema de gestión del tiempo y elaboración de informes que permaneció en producción durante años sin modificaciones importantes.

Análisis e informes periódicos de progreso

Entregué análisis e informes periódicos al CEO, convirtiendo los KPI de ingeniería y el estado de las entregas en información para tomar decisiones.

Primer panel de observabilidad de la empresa

Desarrollé y puse en marcha el primer panel de observabilidad de la empresa, con información del rendimiento del sistema en tiempo real y visualizaciones de datos en el televisor grande de la oficina.

Comunicación de marca guiada por la respuesta del público

Utilicé investigación de mercado y opiniones del público para orientar la comunicación de marca y las acciones de contacto de Engineer ApS.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Analítica propia dentro de PostgreSQL · Desarrollo de informes de análisis de datos · 4.000 horas ahorradas con CI/CD GitHub y GitLab · Optimización de previsiones e inversiones · 500 informes de datos eléctricos y GIS · 8 proyectos Power BI con manuales · 30 días de datos eléctricos en 180 ms

{ Datos y bases de datos }

Gobernanza y calidad de datos

Confiar en las cifras

Reglas y comprobaciones que detectan datos incorrectos al entrar y aclaran quién responde de cada cifra y de dónde procede.

13 direcciones medidas sin rastreo

Medí trece direcciones únicamente a partir del registro de accesos del servidor web — sin cookies, rastreadores ni servicios de terceros —, enmascarando la red del cliente al escribir cada línea e incorporando el resultado a un archivo permanente de 21 recuentos diarios y 20 facetas mensuales.

Conocer la licencia de todo lo que se publica

Audité la compatibilidad de licencias de 644 crates Rust en cada compilación y probé que el control se activa al cambiar la licencia de una crate y al modificar la revisión fijada del núcleo sin regenerar los datos.

Pruebas para los propios verificadores

Escribí pruebas para los propios verificadores tras constatar que un verificador que solo recibe entradas correctas puede acabar informando de éxito por no haber leído nada; introduje una falta de ortografía para confirmar su detección y obtuve un intervalo de identificadores de la base de datos en vez de fijarlo en la prueba.

Regla de alcance tras una fuga de credenciales

Incorporé al repositorio una regla de alcance tras una reorganización que había introducido el inventario, los permisos de cortafuegos y textos de otra empresa, y mantuve los residuos en cuarentena bajo el detector de secretos, sin excluirlos.

Recopilación y análisis de requisitos de negocio

Recopilé y analicé requisitos de negocio para convertirlos en funcionalidades y en historias de usuario aplicables, de acuerdo con las normas de gobernanza de datos.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Comprobación de cifras entre idiomas · Todas las cadenas en un árbol de contenido · 963 bloques de metadatos por un único grafo · Analítica propia dentro de PostgreSQL · 348 tablas en 34 esquemas Postgres · Combinación recursiva guiada por un catálogo · Datos PostgreSQL centrados en funciones · 600 flujos ETL/ELT en PL/pgSQL · 500 informes de datos eléctricos y GIS · 1.000 incidencias de datos geográficos resueltas · Un plan de infraestructura para 14 departamentos

{ Datos y bases de datos }

Preparación de datos para IA

Datos que la IA puede utilizar de verdad

Sus registros al alcance de los agentes de IA, una única versión de cada cliente, acceso limitado a lo que corresponde y cada respuesta vinculada a su fuente.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

963 bloques de metadatos por un único grafo · 495 páginas generadas desde SQLite · 1.022 migraciones con control de coherencia · 348 tablas en 34 esquemas Postgres · Diseño y operación de 6 flujos ETL/ELT · Flujo de datos geográficos 50 veces más rápido

{ Nube, infraestructura y DevOps }

Infraestructura en la nube y migración

El lugar donde funciona el software

Servidores dimensionados para el uso real, trasladados a la nube o fuera de ella, con la factura mensual calculada de antemano.

Segundo servidor aprovisionado desde código

Automaticé el aprovisionamiento de un segundo servidor en un segundo proveedor de nube y creé el cortafuegos antes que la máquina para que naciera protegida, con los cortafuegos de ambos proveedores configurados directamente mediante sus API REST para evitar una colección de terceros.

Contenedores con Podman y Quadlet

Desplegué una plataforma de contenedores con Podman y Quadlet bajo systemd en lugar de Docker, porque la publicación predeterminada de puertos en redes bridge de Docker puede eludir las reglas UFW, y asigné a los despliegues un usuario sin privilegios con una única orden fija, en vez de root.

Infraestructura definida en un repositorio

Desarrollé la infraestructura de la empresa mediante 19 playbooks de Ansible y 34 roles en 12.065 líneas de YAML, haciendo converger un servidor activo hacia un estado declarado con todas las ejecuciones idempotentes.

Despliegues completos de la plataforma en Azure

Asumí los despliegues completos de la plataforma en Azure y gestioné las versiones en desarrollo, preproducción y producción.

Infraestructura Azure como código

Aprovisioné infraestructura Azure como código con Bicep — Container Apps, PostgreSQL Flexible Server, Front Door/WAF y redes — en los entornos de desarrollo, preproducción y producción.

20 aplicaciones Docker en Google Cloud y AWS

Desplegué y mantuve 20 aplicaciones en contenedores Docker, resolví incidencias con Podman y Kubernetes y administré aplicaciones basadas en R en Google Cloud y AWS.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Servicios públicos en un servidor pequeño · Infraestructura principal creada desde cero · Infraestructura simplificada fuera de AWS

{ Nube, infraestructura y DevOps }

Infraestructura como código

Servidores que se pueden reconstruir a partir de una configuración versionada

Toda la configuración escrita en archivos para poder recrearla con exactitud y evitar que exista solo en la memoria de una persona.

Ensayo del hook CI: 2 defectos detectados

Ensayé el hook CI del servidor de la forja y encontré dos defectos imposibles de alcanzar leyendo el archivo: una alternativa que introducía un argumento irresoluble en la entrada del hook y una variable de entorno de git que acompañaba al control hasta el checkout y hacía fallar 19 pruebas.

Secretos operativos separados por servidor

Separé los cuatro secretos operativos de cada servidor tras constatar que dos servidores que comparten una señal de actividad alertan menos que dos señales independientes, y que una contraseña compartida de copias de seguridad puede exponer las copias de ambos servidores, incluso en repositorios separados, si alguien la obtiene y puede leer los datos cifrados y los archivos de claves de ambos repositorios.

Menor exposición en cada unidad systemd

Reduje la exposición de aislamiento systemd en todas las unidades que instala la plataforma — un servicio de monitorización por ausencia de señal pasó de 9,6 UNSAFE a 1,5 y una forja git expuesta a Internet de 8,3 EXPOSED a 1,5 — y añadí una comprobación del analizador durante la convergencia tras encontrar una directiva mal escrita que se ignoraba silenciosamente en tres plantillas de unidades.

Zona DNS de 20 registros como código

Concilié declarativamente una zona DNS de 20 registros con la API de Cloudflare mediante puntos de entrada separados para auditoría y exportación BIND, y desactivé el proxy CDN por motivos de privacidad después de implementarlo.

Comprobación de secretos en un segundo

Añadí una comprobación previa que ejecuta en aproximadamente un segundo el mismo código de convergencia sobre secretos locales del operador, después de que una ejecución de producción parcialmente aplicada fallara en su novena tarea con los ajustes de intercambio ya escritos en el servidor activo.

Modo de comprobación Ansible fiel a la realidad

Hice que el modo de comprobación reflejara la realidad en toda la plataforma tras detectar seis sondeos que decidían a partir de un valor que el servidor nunca había proporcionado, porque el módulo command de Ansible informa de éxito con --check aunque omita la orden por completo.

Monitorización por ausencia de señal

Desarrollé una monitorización por ausencia de señal que solo avisa de actividad mientras la memoria y el disco están en buen estado, de modo que un servidor degradado genera una alerta al quedar en silencio; además detecté seis variables llamadas «free» cuando el control medía correctamente «available», valores separados por un orden de magnitud en una máquina de 464 MB.

Automatización de copias con un método explícito de restauración

Implementé automatización de copias con restic, retención, comprobaciones de integridad y un flujo mensual de prueba de restauración; documenté la activación del almacenamiento y la verificación de restauración real como pasos operativos separados.

SSH reforzado con 24 directivas verificadas

Reforcé SSH con 24 directivas verificadas mediante validación en tres etapas — archivo candidato, configuración ensamblada y lectura de la configuración por el propio demonio —, después de que esta última detectara que el servidor en ejecución anulaba silenciosamente dos de las veinticuatro.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Espejo onion de Tor con una dirección legible · Segundo servidor aprovisionado desde código · Su código en su propio servidor · Servicios públicos en un servidor pequeño · Infraestructura definida en un repositorio · Copias y recuperación ante desastres como código · Infraestructura Azure como código · 1.000 reglas RBAC para servidores Linux

{ Nube, infraestructura y DevOps }

DevOps y automatización CI/CD

Publicar cambios sin sobresaltos

Cambios probados, publicados y reversibles, para que lleguen a sus clientes un martes por la tarde en vez de un viernes por la noche.

Un criterio de historial para 4 repositorios

Apliqué un único criterio de historial a cuatro repositorios — commits convencionales, sin emojis ni líneas de atribución y con un hook de mensajes de commit —, junto con 48 documentos de instrucciones que regulan la forma de trabajar.

Pruebas reducidas a 8,9 segundos

Pasé de seis pruebas que superaban un límite de sesenta segundos a 135 pruebas satisfactorias en 8,9 segundos al analizar el hilo principal y eliminar las dos llamadas en las que permanecía durante 3.989 de 4.017 muestras.

Diferencias detectadas entre hooks y controles

Detecté que los hooks de commit y el control de calidad ejecutaban comprobaciones distintas aunque un documento prometía que eran iguales, al comparar ambas listas en una prueba: las cinco que solo se ejecutaban a mano eran las que revisaban la redacción del CV.

Control de commits con 22 analizadores

Desarrollé un control obligatorio de commits con 22 analizadores de una línea y cinco que requieren una explicación, sin nivel de advertencias ni supresiones en línea permitidas, que cubre HTML, CSS, JavaScript, Python, YAML, Markdown, Shell, enlaces, ortografía, secretos y tipografía.

Análisis continuo de seguridad

Establecí un programa continuo de seguridad — análisis de código, DAST, comprobación de dependencias y vulnerabilidades, generación de SBOM, detección de secretos y acciones fijadas por SHA —, junto con 21 auditorías de seguridad documentadas.

Motor de 268 controles para el repositorio

Desarrollé el motor de comprobaciones del repositorio — 268 controles de commit registrados, 277 reglas de análisis estático y 15 reglas ESLint propias —, además de 146 pruebas de los propios controles, para que la compilación haga cumplir el criterio de calidad.

Colaboración del equipo automatizada

Automaticé la colaboración del equipo, la gestión de contraseñas, tareas y tiempo, y desarrollé un sistema semiautomático de presentación de proyectos que elevó la productividad del equipo.

Cero advertencias en seis lenguajes

Fijé un criterio de calidad de cero advertencias en seis lenguajes — Go, TypeScript, SQL, Python, Shell y Markdown — aplicado mediante hooks de pre-commit.

578 tareas de automatización go-task

Redacté 578 tareas de automatización go-task para los modos de desarrollo nativo, Docker y HTTPS, así como para análisis estático, pruebas, bases de datos y despliegue.

Flujos CI/CD en GitHub Actions

Desarrollé flujos CI/CD en GitHub Actions con una imagen de producción distroless para la interfaz y promoción entre varios entornos.

CI/CD optimizado: 4.000 horas ahorradas

Simplifiqué los procesos CI/CD y ahorré 4.000 horas mediante la automatización de los flujos de desarrollo de software.

SSL/TLS automático para 100 aplicaciones Docker

Automaticé la creación de certificados SSL/TLS para 100 aplicaciones Docker, manteniendo conexiones seguras entre servidores Ubuntu Linux.

Menor riesgo con Grafana y Prometheus

Reduje los riesgos operativos mediante un panel de monitorización con Grafana y Prometheus, mejorando la fiabilidad del sistema.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Conocer la licencia de todo lo que se publica · La mitad sin utilizar del núcleo, implementada · Analizador que verifica cada llamada FFI · Ensayo del hook CI: 2 defectos detectados · Pruebas para los propios verificadores · Cada regla del analizador tratada como error · Generador: 981 pruebas y 92 % de cobertura · Control de calidad de 10.242 líneas analizado · Sitemap corregido: fechas que eran iguales · Espejos del sitio en Gemini y Gopher · Una red de seguridad que ya no cuesta media hora · Contenedores con Podman y Quadlet · Su código en su propio servidor · Comprobación de secretos en un segundo · Modo de comprobación Ansible fiel a la realidad · Infraestructura definida en un repositorio · 1.022 migraciones con control de coherencia · Monitorización de errores y trazas propias · Pruebas para Go, SQL y navegador · Infraestructura principal creada desde cero · Despliegues completos de la plataforma en Azure · Contrato de API generado desde la base de datos · Infraestructura simplificada fuera de AWS · Datos automatizados con Python y PL/pgSQL · 4.000 horas ahorradas con CI/CD GitHub y GitLab · Procesos renovados: 8.000 horas ahorradas · 20 aplicaciones Docker en Google Cloud y AWS · 100 copias críticas automatizadas con Barman · Entrega automática de 20 flujos de datos GIS · Despliegue GIS SaaS con GitHub Actions

{ Nube, infraestructura y DevOps }

Contenedores y orquestación

Software que funciona igual en todas partes

Software empaquetado para comportarse igual en un portátil, un servidor de pruebas y producción, y ampliar su capacidad cuando aumente la actividad.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Contenedores con Podman y Quadlet · Menor exposición en cada unidad systemd · Flujos CI/CD en GitHub Actions · SSL/TLS automático para 100 aplicaciones Docker · 20 aplicaciones Docker en Google Cloud y AWS

{ Nube, infraestructura y DevOps }

Fiabilidad de sistemas y monitorización

Saberlo antes que sus clientes

Monitorización y alertas que avisan cuando un fallo aún es pequeño, y cifras que muestran con qué frecuencia se mantiene así.

Seis recolectores declaran lo que omitieron

Hice que seis recolectores de informes declararan la parte de sus entradas que nunca habían leído — archivos no abiertos, sondeos no instalados y líneas del diario no reconocidas — y ajusté los umbrales a 116 muestras medidas tras constatar que una recolección ciega y un servidor sano generan la misma página.

7 roles de informes de solo lectura

Desarrollé siete roles de informes de solo lectura que representan un servidor activo en Markdown — inventario técnico, acceso, git, métricas, tráfico, seguridad e inventario de proveedores — bajo la regla de no imprimir ninguna cifra que la ejecución no haya medido.

Bloqueo de intrusiones probado en cada ejecución

Verifiqué de extremo a extremo el bloqueo de intrusiones en cada ejecución de refuerzo, bloqueando una dirección reservada para pruebas, comprobando la regla resultante del núcleo y retirando el bloqueo en un bloque de limpieza garantizado, para que una jail averiada haga fallar la ejecución en vez de presentarse como operativa.

Tres cerraduras en la puerta, ninguna cerrada

Detecté y corregí tres protecciones contra fuerza bruta SSH que nunca habían funcionado: una jail de bloqueo que vigilaba el puerto 22 mientras el demonio escuchaba en el 1986, un límite de tasa oculto bajo una regla más amplia y una acción de bloqueo cuyo binario nunca se encontraba, por lo que no se había aplicado ningún bloqueo.

Transmisión IPTV ininterrumpida

Mantuve la entrega ininterrumpida de señales IPTV entre proveedores y clientes, monitorizando y manteniendo la red de distribución y la telefonía IP las 24 horas.

Soporte 24/7 para unos 1.000 servidores IPTV/OTT

Presté soporte de infraestructura 24/7 a una plataforma de distribución IPTV/OTT, administrando unos 1.000 servidores y sistemas de los clientes para usuarios de China, Estados Unidos y Alemania.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

13 direcciones medidas sin rastreo · Un flujo de eventos limitado para una aplicación macOS · Secretos operativos separados por servidor · Modo de comprobación Ansible fiel a la realidad · Monitorización por ausencia de señal · Automatización de copias con un método explícito de restauración · Límites de tasa y protección contra abusos · Monitorización de errores y trazas propias · Trabajo lento trasladado a una cola River · Infraestructura aún en servicio años después · Despliegues completos de la plataforma en Azure · Correo con conmutación y registro de entregas · Un producto cartográfico de principio a fin · 40 sitios web con Apache y Nginx · Menor riesgo con Grafana y Prometheus · 10 servidores PostgreSQL y MS SQL en Ubuntu · Primer panel de observabilidad de la empresa

{ Nube, infraestructura y DevOps }

Copias de seguridad y recuperación ante desastres

Recuperarlo todo después de un mal día

Copias de seguridad que se prueban en vez de darse por buenas y un plan escrito para la mañana en que un servidor no vuelva a encenderse.

Administración de 30 VPS Ubuntu Linux

Administré 30 instancias VPS con Ubuntu Linux, implementé estrategias de recuperación ante desastres y mantuve configuraciones de red adecuadas.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Secretos operativos separados por servidor ·

Automatización de copias con un método explícito de restauración ·

Copias y recuperación ante desastres como código · 100 copias críticas automatizadas con Barman

{ Nube, infraestructura y DevOps }

Administración de sistemas

Mantenerlo todo en funcionamiento

El cuidado diario de servidores, correo, cuentas y actualizaciones para que los servicios de los que depende todo el mundo sigan funcionando sin sorpresas.

Reglas de cortafuegos verificadas por posición

Verifiqué las reglas del cortafuegos por su posición, además de por su existencia, leyendo la lista numerada de reglas y la cadena activa del filtro de paquetes, porque una regla presente puede no recibir nunca un paquete.

Gestor de contraseñas para toda la empresa

Integré un sistema de gestión de contraseñas para toda la empresa, reforzando la seguridad y simplificando el control de accesos.

Administración de 100 servidores Bare Bone

Administré 100 servidores Bare Bone, redes físicas y sistemas de telefonía IP, proporcionando una infraestructura sólida para el crecimiento de la empresa.

Configuración y despliegue de 1.000 routers Wi-Fi

Configuré y desplegué 1.000 routers Wi-Fi, mejorando el acceso a la red y su rendimiento para los clientes.

Administración de infraestructura de red

Administré la infraestructura de red de más de 1.000 servidores, atendiendo al despliegue, la seguridad y la resolución de incidencias de los sistemas.

Gestión de 8 conexiones VPN WireGuard e IPsec

Administré y resolví incidencias en 8 conexiones VPN de WireGuard e IPsec, manteniendo comunicaciones seguras entre Google Cloud y sistemas Linux.

Gestión de accesos con M365 y SSO

Prevení brechas de seguridad al dirigir iniciativas de gestión de accesos con M365, 1Password, Red Hat SSO y OKTA SSO.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Espejo onion de Tor con una dirección legible · Seis recolectores declaran lo que omitieron · Ensayo del hook CI: 2 defectos detectados · Su código en su propio servidor · 7 roles de informes de solo lectura · Menor exposición en cada unidad systemd · Zona DNS de 20 registros como código · Monitorización por ausencia de señal · Automatización de copias con un método explícito de restauración · Bloqueo de intrusiones probado en cada ejecución · SSH reforzado con 24 directivas verificadas · Servicios públicos en un servidor pequeño · Infraestructura definida en un repositorio · Infraestructura principal creada desde cero · Infraestructura aún en servicio años después · Transmisión IPTV ininterrumpida · Soporte 24/7 para unos 1.000 servidores IPTV/OTT · 40 sitios web con Apache y Nginx · Administración de 30 VPS Ubuntu Linux · 1.000 reglas RBAC para servidores Linux · 10 servidores PostgreSQL y MS SQL en Ubuntu

{ Nube, infraestructura y DevOps }

Configuración de redes y VPN

Conectar sus oficinas y a su personal

Wi-Fi, cortafuegos y acceso remoto configurados para trabajar desde cualquier lugar y permitir el acceso solo a las personas autorizadas.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Segundo servidor aprovisionado desde código · Zona DNS de 20 registros como código · Reglas de cortafuegos verificadas por posición · Tres cerraduras en la puerta, ninguna cerrada · Transmisión IPTV ininterrumpida · Infraestructura Azure como código · Administración de 100 servidores Bare Bone · Configuración y despliegue de 1.000 routers Wi-Fi · Administración de infraestructura de red · Gestión de 8 conexiones VPN WireGuard e IPsec · Administración de 30 VPS Ubuntu Linux

{ Nube, infraestructura y DevOps }

Seguridad y gestión de accesos

Decidir quién puede acceder a qué

Cuentas, contraseñas y permisos organizados para que cada persona tenga exactamente el acceso que necesita y lo pierda el mismo día que deja la empresa.

Mejor comunicación y colaboración del equipo

Mejoré la comunicación y la colaboración del equipo mediante Slack, Mattermost, 1Password y Jira, ahorrando 8.000 horas de trabajo.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Espejo onion de Tor con una dirección legible · 13 direcciones medidas sin rastreo · Conocer la licencia de todo lo que se publica · La mitad sin utilizar del núcleo, implementada · Secretos operativos separados por servidor · Regla de alcance tras una fuga de credenciales · Segundo servidor aprovisionado desde código · Contenedores con Podman y Quadlet · Menor exposición en cada unidad systemd · Zona DNS de 20 registros como código · Comprobación de secretos en un segundo · Reglas de cortafuegos verificadas por posición · Bloqueo de intrusiones probado en cada ejecución · SSH reforzado con 24 directivas verificadas · Tres cerraduras en la puerta, ninguna cerrada · Análisis continuo de seguridad · Reclamaciones y decisiones de titularidad · Límites de tasa y protección contra abusos · Stripe y acceso por compras en aplicaciones · Gestor de contraseñas para toda la empresa · Colaboración del equipo automatizada · Aplicación reforzada con CSP basada en nonces · Infraestructura Azure como código · Correo con conmutación y registro de entregas · Cambio de contexto entre organizaciones · SSL/TLS automático para 100 aplicaciones Docker · Administración de infraestructura de red · Gestión de 8 conexiones VPN WireGuard e IPsec · Gestión de accesos con M365 y SSO · 1.000 reglas RBAC para servidores Linux

{ Nube, infraestructura y DevOps }

Seguridad de agentes de IA

Que su IA actúe solo donde le corresponde

El acceso de cada agente definido y limitado a lo necesario, cada acción registrada donde el agente no puede alterarla y cada protección comprobada en lugar de darse por segura.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Secretos operativos separados por servidor · Regla de alcance tras una fuga de credenciales · Comprobación de secretos en un segundo · Bloqueo de intrusiones probado en cada ejecución · Tres cerraduras en la puerta, ninguna cerrada · Análisis continuo de seguridad · Límites de tasa y protección contra abusos · Gestión de accesos con M365 y SSO · 1.000 reglas RBAC para servidores Linux

{ Consultoría y dirección }

Dirección técnica y consultoría

Una ingeniera sénior en la reunión

Alguien de su lado de la mesa cuando una decisión sobre proveedores, presupuesto o arquitectura necesita una segunda opinión informada.

Analizador propio para la documentación

Sometí la documentación a un analizador propio — límite de líneas por archivo que solo puede reducirse, índice obligatorio y comprobación de rutas reales — que detectó seis de nueve planes marcados como terminados con tareas pendientes y catorce tareas obligatorias para commits que no figuraban en ningún documento.

Dirección del desarrollo de proyectos

Guié la ejecución de numerosos proyectos de la empresa y ofrecí asesoramiento especializado durante las fases de desarrollo de software.

Crecimiento de la empresa de 4 a 14 empleados

Dirigí el crecimiento de la empresa de 4 a 14 empleados mediante metodologías ágiles y prácticas eficaces de gestión de proyectos.

50.000 palabras de documentación de software

Redacté 50.000 palabras de documentación completa del software con Markdown en GitHub, Craft y Confluence, preservando el conocimiento y la transparencia del proceso.

Ahorro estimado de 4.000 horas, equipo de 2 a 18

Estimé un ahorro de 4.000 horas mientras ejercía la mentoría de un equipo que pasó de 2 a 18 integrantes, mejorando los flujos de trabajo y la coordinación entre departamentos.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Un criterio de historial para 4 repositorios · Pruebas reducidas a 8,9 segundos · Concurrencia estricta Swift 6 sobre C · Diferencias detectadas entre hooks y controles · Pruebas para los propios verificadores · Cada regla del analizador tratada como error · Generador: 981 pruebas y 92 % de cobertura · Control de calidad de 10.242 líneas analizado · Desarrollo organizado en 51 iniciativas escritas · Una red de seguridad que ya no cuesta media hora · Motor de 268 controles para el repositorio · Pruebas para Go, SQL y navegador · Análisis e informes periódicos de progreso · Decisiones técnicas fundacionales de plataforma · Cero advertencias en seis lenguajes · CI/CD optimizado: 4.000 horas ahorradas · Procesos renovados: 8.000 horas ahorradas · Más de 30 proyectos GIS: desarrollo y soporte · Dirección del desarrollo integral de mapas GIS · Ingresos multiplicados por 10 con mapas GIS · Un plan de infraestructura para 14 departamentos

{ Consultoría y dirección }

Desarrollo de equipos y mentoría

Hacer crecer el equipo que ya tiene

Selección, incorporación y acompañamiento para que el conocimiento se quede con su personal en vez de marcharse con una consultora.

Dirección de un equipo de soporte informático

Dirigí un equipo que ofrecía soporte informático, recuperación de datos y reparación de equipos a más de 1.000 clientes, manteniendo la calidad del servicio.

150 horas al mes ahorradas en 30 proyectos

Mejoré la eficiencia de 30 proyectos y ahorré 150 horas al mes al optimizar los flujos de trabajo y la gestión de recursos.

Creación de un departamento de soporte técnico

Creé un departamento de soporte técnico que atendió a más de 10.000 clientes con servicios de soporte informático y resolución de incidencias, y registré las solicitudes en un CRM.

Gestión de 5.000 horas de desarrollo de mapas

Diseñé y gestioné 5.000 horas de desarrollo de mapas con Scrum y Jira para organizar los proyectos.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Dirección del desarrollo de proyectos · Mejor comunicación y colaboración del equipo · Crecimiento de la empresa de 4 a 14 empleados · Entrega de 30 proyectos GIS satisfactorios · Ahorro estimado de 4.000 horas, equipo de 2 a 18

{ Consultoría y dirección }

Gestión ágil de proyectos

Llevar el proyecto a término

Una persona que responda del alcance, las fechas y las conversaciones difíciles, para que el proyecto tenga una responsable en lugar de cinco opiniones.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Un criterio de historial para 4 repositorios · Analizador propio para la documentación · Desarrollo organizado en 51 iniciativas escritas · Colaboración del equipo automatizada · Sistema de gestión del tiempo e informes · Análisis e informes periódicos de progreso · 150 horas al mes ahorradas en 30 proyectos · Creación de un departamento de soporte técnico · Procesos renovados: 8.000 horas ahorradas · Dirección del desarrollo de proyectos · Mejor comunicación y colaboración del equipo · Crecimiento de la empresa de 4 a 14 empleados · Entrega de 30 proyectos GIS satisfactorios · Recopilación y análisis de requisitos de negocio · Gestión de 5.000 horas de desarrollo de mapas · Ahorro estimado de 4.000 horas, equipo de 2 a 18

{ Consultoría y dirección }

Estrategia de producto y requisitos

Decidir qué construir

Convertir las necesidades del negocio en algo que se pueda desarrollar y decir con claridad qué no merece la pena construir.

Una identidad coherente entre canales

Desarrollé la identidad de marca de Engineer ApS en materiales físicos y digitales, vinculando su presentación con su portafolio de ingeniería. La fundadora informa de un aumento de 200 veces en el reconocimiento de marca gracias a este trabajo.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Errores que dejan de culpar al usuario · Desarrollo organizado en 51 iniciativas escritas · Control de commits con 22 analizadores · Espacios de contratación y carrera marítima · Capa social: publicaciones, noticias y grupos · Monedero de fidelidad, ligas e insignias · Reclamaciones y decisiones de titularidad · Panel interno de operaciones de la plataforma · Stripe y acceso por compras en aplicaciones · Sistema de gestión del tiempo e informes · Diseño integral de UI y UX del producto · Desarrollo de informes de análisis de datos · Recopilación y análisis de requisitos de negocio · Gestión de 5.000 horas de desarrollo de mapas · 200 mejoras de UI/UX presentadas · Optimización de previsiones e inversiones · Ingresos multiplicados por 10 con mapas GIS · Comunicación de marca guiada por la respuesta del público

{ Consultoría y dirección }

Documentación técnica

Dejarlo escrito para quien venga después

Traspasos, procedimientos operativos y manuales para que su personal pueda operar el sistema sin llamar a quien lo construyó.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Seis recolectores declaran lo que omitieron · Un criterio de historial para 4 repositorios · Errores que dejan de culpar al usuario · Analizador que verifica cada llamada FFI · Comprobación de cifras entre idiomas · Diferencias detectadas entre hooks y controles · 106 de 106 comprobaciones del perfil PDF/UA-1 · Analizador propio para la documentación · Desarrollo organizado en 51 iniciativas escritas · Control de commits con 22 analizadores · Regla de alcance tras una fuga de credenciales · 7 roles de informes de solo lectura · Análisis continuo de seguridad · Motor de 268 controles para el repositorio · 578 tareas de automatización go-task · API migrada a Huma v2 con paridad del 100 % · 50.000 palabras de documentación de software · 8 proyectos Power BI con manuales · Un plan de infraestructura para 14 departamentos

{ Soporte de TI y equipos }

Soporte informático y asistencia técnica

Ayuda informática para el día a día

Resolver con rapidez y lenguaje claro los problemas del portátil, la impresora, el acceso o el correo, sin hacer sentir mal a nadie.

Diagnóstico y reparación de portátiles

Diagnostiqué y reparé hardware de portátiles — pantallas, bisagras, teclados, paneles táctiles, placas base y alimentación — y resolví problemas de software en Linux, Windows y Mac.

Datos recuperados de SD, HDD, SSD y RAID

Recuperé datos de una amplia variedad de medios — tarjetas SD, HDD, SSD, matrices RAID, unidades externas y sistemas Mac.

Gestión de 4.000 reparaciones de ordenadores

Gestioné 4.000 reparaciones de ordenadores para resolver con rapidez y eficacia problemas de hardware y software.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Soporte 24/7 para unos 1.000 servidores IPTV/OTT · Configuración y despliegue de 1.000 routers Wi-Fi · Dirección de un equipo de soporte informático · Creación de un departamento de soporte técnico

{ Soporte de TI y equipos }

Reparación de equipos y recuperación de datos

Rescatar archivos de un equipo averiado

Reparar lo que merece la pena y recuperar fotografías,
facturas o planos de un disco que ha dejado de funcionar.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Diagnóstico y reparación de portátiles · Datos recuperados de SD, HDD, SSD y RAID · Gestión de 4.000 reparaciones de ordenadores · Dirección de un equipo de soporte informático

{ Marca y marketing }

Marca, marketing y SEO

Que le encuentren y reconozcan su identidad

Un nombre, una imagen y un sitio web que aparezcan al buscar
lo que vende y transmitan lo mismo allí donde esté presente.

Este servicio forma parte de trabajos que figuran en otros lugares de esta página:

Identidad visual creada a partir de 2 dibujos · Sistema de diseño macOS de 48 tokens · Sitemap corregido: fechas que eran iguales · Paquete CSS reducido de 87 KB a 31 KB · 963 bloques de metadatos por un único grafo · Sistema de color cerrado en 28 colores · 495 páginas generadas desde SQLite · 27 tamaños de fuente por una escala de 6 pasos · Desarrollo web combinado con SEO y contenidos · Comunicación de marca guiada por la respuesta del público · Una identidad coherente entre canales

Compare su punto de partida

Elija un trabajo comparable e identifique qué cambia en su propia situación.

Vea cuánto cuesta Póngase en contacto →

Hablemos de un enfoque relevante

Pídanos que convirtamos la experiencia relevante en un enfoque que pueda evaluar.

Explore nuestros servicios

welcome@engineer.company +45 50 10 61 84 Copenhague, Dinamarca

Llévese esta página

Tarjeta de contacto

Creado con esmero y cariño por D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Derechos de autor © 2025 – actualidad · Engineer ApS · Todos los derechos reservados.

Impreso desde https://engineer.company/es/portfolio/