Engineer

Portefølje

Ingeniørydelser med eksempler på udført arbejde.

Engineer ApS blev registreret i 2025. Det meste af arbejdet her blev leveret inden da af vores stifter i kundernes egne teams; noget er leveret af virksomheden selv, og noget senere i et ansættelsesforhold. De samme hænder og de samme standarder — vi siger, hvad der er hvad, i stedet for at sløre det.

Læs om arbejdet **i sin sammenhæng** — beslutningerne, leverancen og de beskrevne resultater.

Se hele kataloget

Find erfaring, der er værd at bruge

Brug oversigten til at finde erfaring, der er relevant for jeres projekt.

Se vores ydelser Kontakt os →

Læs beslutningerne bag arbejdet

Eksemplerne beskriver leveret arbejde på tværs af ydelser. Læs en gennemgang for at se situationen og beslutningerne.

Cases Kontakt os →

Disse navne **går igen i arbejdet nedenfor** — intet eksotisk, som jeres team skal overtage.

Python PostgreSQL TypeScript React GitHub Docker

Se alle teknologierne

Se efter kategori

SOFTWAREUDVIKLING

Backend-udvikling Frontend-udvikling Full stack-udvikling API'er & integration Webudvikling Designsystemer & UI UX/UI-design Test & QA Optimering af ydeevne Python Internationalisering

DATA & DATABASER

Databaser · PostgreSQL · SQL · Data engineering · Datapipelines (ETL/ELT) · Dataanalyse · Data governance · Data warehousing · GIS / geodata

CLOUD, INFRASTRUKTUR & DEVOPS

Cloud · Containere (Docker/Kubernetes) · DevOps · Automatisering & CI/CD · Infrastruktur · Linux & servere · Systemadministration · Netværk & VPN · Overvågning & observerbarhed · Driftssikkerhed & backup · Sikkerhed · Platformarkitektur · Migrering & modernisering

RÅDGIVNING & LEDELSE

Teknisk ledelse · Teamledelse · Mentoring & coaching · Agile & Scrum · Projektledelse · Produkt & krav · Interessenter & rapportering · Dokumentation · Løsningsarkitektur · Brand & marketing

Alt herunder **er leveret** — ikke foreslået, ikke planlagt.

Hver post fortæller, hvad arbejdet ændrede, og linker til de ydelser, den demonstrerer.

Det, vi tilbyder Hvad det koster →

{ Softwareudvikling }

Platform- & løsningsarkitektur

At beslutte, hvordan det hele skal bygges

Systemets form lagt fast på papir først, så delene passer sammen, og prisen er kendt, før nogen skriver kode.

Har indført Swift 6 strict concurrency over C

Har indført Swift 6's complete strict concurrency uden actors og bygget bro fra en blokerende C-hændelsesløkke til main actor gennem én producent, én forbruger og én rækkefølge — efter at have fastslået, at en opgave pr. hændelse mister den rækkefølge, grænsefladen afhænger af.

Et afgrænset hændelsesflow i en macOS-app

Har begrænset hændelsesbuffere og logning efter en registreret macOS-hændelse med vækst på 41 MB i sekundet og 111 GB komprimerede sider på en maskine med 36 GB; en registreret fuld testkørsel reducerede kernens loglinjer fra 610.996 til 1.411.

Har bygget en native macOS-beskedklient i Swift 6

Har bygget en native macOS-beskedklient i Swift 6 og SwiftUI — 11.141 linjer fordelt på 53 filer — oven på C-grænsefladen til en Rust-kerne linket som et statisk arkiv fra en fastlåst revision.

Har indlejret en QR-encoder på 522 linjer

Har indlejret en QR-encoder — Reed-Solomon over GF(256), fast modullayout, otte maskemønstre — på 522 linjer i stedet for at tage en tredje runtime-afhængighed ind og verificeret den ved at læse den færdige matrix tilbage med en uafhængigt skrevet dekoder.

Har bygget en databasedrevet CV-generator

Har bygget en databasedrevet generator til CV, referencer, portefølje og ansøgninger i Python — 41 moduler, 10.580 linjer — der genererer seks outputformater fra én SQLite-kilde med 23 tabeller, som en idempotent pipeline i 19 trin bygger op.

Offentlige tjenester på en lille server

Har drevet virksomhedens offentlige tjenester på en vært med 512 MB og én kerne — en git-forge, en webserver til syv domæner, Tor, to servere til alternative protokoller og blokering af indtrængningsforsøg — med 464 MB brugbar hukommelse som designbegrænsning; backupautomatik er en særskilt implementering.

Har bygget hele kerneinfrastrukturen fra bunden

Har som en af de første medarbejdere designet og bygget hele kerneinfrastrukturen og de understøttende processer fra bunden for en SaaS-startup inden for grøn energi og lagt fundamentet for hurtig vækst.

Har bygget infrastruktur, der holder i årevis

Har planlagt og implementeret ny infrastrukturfunktionalitet til interne og eksterne systemer og bygget løsninger, der er så robuste, at de stadig kører år senere med minimale ændringer.

Har fastlagt platformens grundbeslutninger

Har fastlagt platformens grundlæggende beslutninger i ugerne efter, at repositoryet blev oprettet i november 2025 — lagdelingen, database-first-dataadgangen og standarden med nul advarsler — og de holder stadig ni måneder senere.

Har modelleret 348 tabeller i 34 Postgres-skemaer

Har modelleret det maritime domæne i 348 normaliserede tabeller fordelt på 34 PostgreSQL-skemaer — fagfolk, virksomheder, skibe, jobs, anmeldelser og resten — med SMALLINT-opslagstabeller og UUID v7-nøgler.

Har designet et funktionsbaseret Postgres-datalag

Har designet et funktionsbaseret datalag i PostgreSQL — 1.275 lagrede funktioner fordelt på 34 skemaer — så al læsning og skrivning går gennem en funktion, databasen kan tildele rettigheder til, og ikke direkte gennem en tabel.

Har migreret API'et til Huma v2 med 100 % paritet

Har migreret HTTP-API'et fra Fiber til Huma v2 — 649 paths og 760 operationer — og opnået og fastholdt 100 % paritet mellem de ruter, serveren registrerer, og den OpenAPI-beskrivelse, den udgiver.

Har bygget skift af organisationskontekst

Har designet et system til skift af organisationskontekst med localStorage i klienten og en spejlet cookie på serveren, så brugere kan optræde som de organisationer, de administrerer, mens autorisationen følger mindste privilegium.

Har designet en JSON-passthrough-arkitektur

Har designet en JSON passthrough-arkitektur, hvor PostgreSQL-funktioner returnerer komplet JSON, som Go-API'et videresender uændret, så der ikke er nogen mellemliggende unmarshalling, og frontenden er afkoblet fra skemaændringer.

En maritim platform med klare systemgrænser

Har designet arkitekturen for en lagdelt maritim platform, der adskiller en Next.js PWA-frontend, et Go (Huma/Fiber) API og et PostgreSQL-funktionslag og holder al forretningslogik i databasen.

Forenklet infrastruktur uden for AWS

Har omlagt en international kundes infrastruktur ved at fjerne ubrugte tjenester, tilpasse kapaciteten og flytte arbejdsbelastninger væk fra AWS. Grundlæggeren beskriver en reduktion af skyomkostningerne med en faktor 10 uden tab af produktivitet.

Et kortprodukt, båret fra ende til anden

Har designet, udviklet og understøttet infrastruktur, databehandling og en kortapplikation over 2 år med både softwareleverancer og driftsvedligeholdelse.

Har fornyet processer og sparet 8.000 timer

Har gennemgribende fornyet interne processer og sparet 8.000 timer ved at forbedre softwarearkitektur, systemer og planlægningseffektivitet.

Har ført en GIS-kortapp til 10x omsætningsvækst

Har ledet softwareudviklingen af en GIS-kortapplikation, øget omsætningen 10 gange og positioneret produktet som et primært dataaktiv.

Én infrastrukturplan til 14 afdelinger

Har designet en omfattende infrastrukturramme for et forskningsinstitut, der omfatter 14 afdelinger, med fleksible moduler, samlede datapipelines og strukturerede supportstrategier, med henblik på langsigtet forankring.

{ Softwareudvikling }

Backend- & API-udvikling

Motoren bag jeres app eller website

Den del, brugerne aldrig ser: hvor data ligger, hvilke regler der gælder, og hvordan andre systemer må tale med den.

Har taget den ubrugte halvdel af kernen i brug

Har taget den halvdel af beskedkernen i brug, som applikationen aldrig havde brugt — backupoverførsel, forsvindende beskeder, redigering og gensendelse af beskeder, verificerede invitationer, proxyer og krypteringspolitik — og kørt hver test mod det rigtige bibliotek uden mocks.

Har skrevet en parser, der tjekker hvert FFI-kald

Har skrevet en parser, der læser den rigtige C-header på 7.308 linjer og verificerer hvert kaldssted, hver enum-konstant, og at hver klasse, der ejer en pointer, er final, efter at en håndskrevet pladsholder-header lod kald til tre fjernede funktioner kompilere, linke og crashe.

Har flyttet alle tekststrenge til et indholdstræ

Har flyttet hver streng, brugeren ser, ud af Python og ind i et indholdstræ på 874 filer på tre sprog, efter at have fundet døde oversættelser, som ingen kunne se var døde, og et tjek, der ubemærket kun bedømte en tredjedel af resultaterne.

Har slået alle linterregler til som fejl

Har slået hver eneste regel i Python-linteren til som fejl og arbejdet 1.815 fund ned til nul, hvor hver af de få undtagelser har en skriftlig begrundelse, og to af dem er sikret af et tjek i stedet for en kommentar.

Har givet generatoren 981 tests og 92 % dækning

Har holdt generatoren på 981 testtilfælde og en nedre grænse på 92 % grendækning med advarsler behandlet som fejl og eftervist idempotens ved at køre hele build-pipelinen to gange fra en tom fil og kræve, at det andet gennemløb intet ændrer.

Har bygget job- og karriereværktøj til søfarende

Har bygget job- og karriereværktøjer til søfarende; det registrerede implementeringsøjebliksbillede omfatter 151 lagrede funktioner i 48 tabeller til stillinger, ansøgninger, certifikater, rangudvikling og sejltid.

Har bygget det sociale lag: opslag, feed, grupper

Har bygget platformens sociale lag — opslag, feed, grupper, mentions, en følgergraf og et digest over mærkedage — på det samme funktionsbaserede datalag som resten af produktet.

Har bygget loyalitetspoint, ligaer og badges

Har bygget loyalitets- og omdømmesystemet — 67 funktioner over en hovedbog på 31 tabeller, med ligaer, badges og en indløsningsshop — og med en rækkelås på saldoen til at lukke vinduet for dobbeltforbrug.

Har bygget fail-closed rate limiting mod misbrug

Har bygget fail-closed misbrugskontroller — 22 Redis-baserede rate limiters, Cloudflare Turnstile, idempotens på requests og en origin-lås — så platformen afviser bots og flood-angreb i stedet for at stole på dem, der kalder den.

Har bygget egen fejlovervågning og tracing

Har bygget egen fejlovervågning og OpenTelemetry-tracing i stedet for at købe dem — sanitering af payloads, detektion af spikes og regressioner, symbolication og et syntetisk heartbeat — bag 11 operatørvisninger.

Har flyttet langsomme opgaver til en River-jobkø

Har flyttet langsomt arbejde væk fra request-stien og over på en River-jobkø — 15 worker-moduler, 8 planlagte opgaver og 20 pg_cron-jobs — så en request returnerer, mens arbejdet bag den kører videre.

Har bygget adgangsret via Stripe og in-app-køb

Har bygget betalings- og rettighedslaget — Stripe side om side med app-køb via Apple og Google — så adgangen til kataloget, søgningen og eksporten styres af en adgangsmodel på 11 tabeller, der kontrolleres på serveren.

Har bygget en testsuite til Go, SQL og browseren

Har bygget en lagdelt automatiseret testsuite — 981 Go-tests, 543 frontend- og browserspecs, 494 SQL-adfærdstests — med mutationstest, property-based tests og en tilgængelighedsgate.

Har bygget og vedligeholdt Django-apps til IPTV

Har udviklet og vedligeholdt Django-webapplikationer til IPTV-platformen, udgivet nye funktioner og forbedret ydeevnen og stabiliteten.

Har hærdet applikationen med nonce-baseret CSP

Har hærdet applikationen med nonce-baseret CSP, HSTS, SameSite-cookies, least-privilege-databaseroller og fornyet kontrol af rettigheder på serveren.

Har implementeret en katalogdrevet deep-merge

Har implementeret en katalogdrevet deep-merge for lagrede JSON-præferencer, hvilket forhindrer nedbrud på grund af manglende nøgler, når skemaet udvikler sig.

Har bygget e-mail med failover og leveringslog

Har bygget e-mail som en funktion i selve platformen — tre udbydere med failover, webhooks for levering, logning af afsendelse og levering, skabeloner og kampagner — sikret af et opstartstjek, så platformen ikke starter uden mindst én udbyder.

Har genereret API-kontrakten ud fra databasen

Har genereret API-kontrakten udad fra databasen — OpenAPI, en typet TypeScript-klient på 44.076 linjer, 61 mock handlers og de grænser, UI'et håndhæver — med en kontrol i hvert led, der fejler ved afvigelser.

Har udviklet et system til analyserapporter

Har udviklet et rapporteringssystem til dataanalyse og øget den kvartalsvise softwareomsætning med 400 % gennem Python-baserede PDF-rapporter.

Har ledet over 30 GIS-projekter

Har ledet udvikling, idriftsættelse og support af over 30 GIS-projekter og udvist ekspertise i PostgreSQL, Bash, Python, JavaScript, GDAL, ArcGIS, PostGIS og Mapbox.

Har styret full-stack-udviklingen af GIS-kort

Har styret full-stack-udviklingen af GIS-kort med ansvar for PostgreSQL, Mapbox, ReactJS og NodeJS for at skabe en integreret løsning.

Har leveret 8 Power BI-projekter med manualer

Har gennemført 8 Power BI-projekter med grundige manualer og integreret Microsoft Power BI-værktøjer med NodeJS API og Python FastAPI til effektiv dataanalyse og visualisering.

Har designet og drevet 6 ETL/ELT-pipelines

Har designet, implementeret og administreret 6 ETL/ELT-pipelines med Google BigQuery, MSSQL, PostgreSQL, shell-scripting, PL/pgSQL og Transact-SQL og integreret data, så et Python-API kan behandle dem effektivt.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har indført Swift 6 strict concurrency over C · Har indlejret en QR-encoder på 522 linjer ·

Har bygget en databasedrevet CV-generator · Har fastlagt platformens grundbeslutninger ·

Har designet et funktionsbaseret Postgres-datalag ·

Har migreret API'et til Huma v2 med 100 % paritet · Har bygget skift af organisationskontekst ·

Har designet en JSON-passthrough-arkitektur · En maritim platform med klare systemgrænser

{ Softwareudvikling }

Frontend-udvikling

De skærme, folk rent faktisk bruger

Brugerflader, der indlæses hurtigt, virker på en telefon og opfører sig ens i alle de browsere, jeres kunder nu bruger.

Har bygget et macOS-designsystem med 48 tokens

Har bygget et designsystem på 48 tokens oven på platformens eget glasmateriale og skrevet den linter, der afviser et magisk tal, en hårdkodet skriftstørrelse eller en animation, der ignorerer præferencen for reduceret bevægelse.

Har navngivet alle 17 ikoner til skærmlæsere

Har navngivet hver eneste ikonknap i grænsefladen for skærmlæsere efter at have opdaget, at send-knappen blev læst op som "arrow up circle, button", og skrevet den linter, der kræver etiketten inden for otte linjer fra ikonet.

Har lintet selve kvalitetsgaten på 10.242 linjer

Har bragt 10.242 linjer kvalitetsgate-JavaScript under en formatter og en linter efter at have fastslået, at det var den største kodemængde i repositoryet og den eneste, som intet værktøj læste, og rettet 13 fund uden at undertrykke nogen.

Har skåret stylesheet-bundtet fra 87 KB til 31 KB

Har skåret stylesheet-bundtet fra 87 KB til 31 KB ved at tage en base64-skrifttype ud af det og fjernet 756 KB fordelt på 22 filer, der blev udgivet ved hver udrulning, men som intet henviste til.

Har bygget et 637 linjer langt printstylesheet

Har bygget et 637 linjer langt printstylesheet, der tager højde for fire dokumenterede adfærdsmønstre i renderingsmotoren, herunder run-in-overskrifter, der blev udskrevet hvidt på hvidt for enhver læser, hvis browser foretrak mørkt.

Har sikret, at reduceret bevægelse virker

Har fået indstillingen for reduceret bevægelse til at virke efter hensigten efter at have fundet elleve selektorer, der stadig animerede trods præferencen, fordi en universel transition-none-regel taber på specificitet til enhver regel med en klasse.

Har fundet et manglende lag i det lyse tema

Har fundet, at det lyse tema havde manglet et fuldsides overlay-lag siden den dag, det blev skrevet, ved at kontrollere, at begge temaer maler hver lagdelt flade med det samme antal lag.

Har låst farvesystemet fast på 28 farver

Har låst farvesystemet fast på 28 dokumenterede farver med en linter, der afviser en værdi, som er malet, men udokumenteret, dokumenteret, men umalet, fejlmålt, skrevet om som en literal eller inden for en perceptuel afstand på 0,02 fra en eksisterende farve.

De tilgængelighedsfejl, jeres tjekværktøj overser

Har fundet og rettet 11 tilgængelighedsfejl, som browseren meldte i orden — otte footer-links, der stadig lå i tab-rækkefølgen bag pointer-events, en scroll-tidslinje, der klippede kolofonen af på tre sider, og en regelmotor, der kørte 70 af sine 105 regler.

Tilgængelighedskontroller på 32 sider i begge temaer

Har implementeret WCAG 2.2 Level AA-kontroller på 32 repræsentative sider i begge farvetemaer med en konformitetsoversigt, kontrol af ét AAA-kriterium og det delvise resultat for kriterium 1.4.13 angivet eksplicit.

Har erstattet 27 skriftstørrelser med 6 trin

Har erstattet 27 gengivne skriftstørrelser, hvis nærmeste naboer lå 0,6 % fra hinanden, med en Major Third-skala i seks trin og skrevet den linter, der afviser en otteogtyvende.

Et sikkerhedsnet, der ikke længere koster en halv time

Har skåret sitets browserdrevne kvalitetsgate fra 1.636 sekunder til 615 ved at afvikle dens tjek med de længste først gennem en worker-pulje begrænset til fire spor, efter at have målt, at alfabetisk rækkefølge kostede 320 sekunder mod 224.

Har bygget et Hugo-site på 3 sprog

Har bygget et flersproget statisk site i Hugo med 108 skabelonfiler, heraf 53 partials, der udgiver hver side i fire repræsentationer fra ét indholdstræ på tre sprog og igen under syv fokuserede subdomæner bygget af det samme træ.

Har bygget platformens operatør-back-office

Har bygget operatørernes back-office — omkring 40 admin-ruter og 128 komponenter, der dækker ejerskabskrav, moderation, feature flags, cache og diagnostik — så platformen kan drives uden databaseadgang.

Har vedligeholdt Hugo og forbedret produktets UI

Har vedligeholdt og forbedret det ældre Hugo-baserede statiske website og bidraget med UI/UX-forbedringer til det primære asset management-produkt.

Har bygget et engelsk/dansk i18n-system

Har etableret et engelsk/dansk internationaliseringssystem med 12.027 beskeder pr. locale fordelt på 458 namespace-filer, et ordforråd, som linteren håndhæver, og et budget på 400 linjer pr. fil.

Har designet hele produktets UI og UX

Har designet produktets UI og UX fra ende til anden — gittervisninger af kataloget, valgfri kort- eller tabelvisning, kravvalidering i realtid og breadcrumb-navigation.

Har skabt et iOS 26 'Liquid Glass'-designsystem

Har designet et iOS 26 'Liquid Glass'-designsystem og en kanonisk komponentoversigt, som lint-regler håndhæver, så brugerfladen forbliver ensartet.

Har leveret en fuld Progressive Web App

Har leveret fuld understøttelse af Progressive Web App — kan installeres og bruges offline — med Workbox runtime-caching via next-pwa.

Har bygget Next.js 16-frontenden: SSR, SSG og CSR

Har bygget Next.js 16-frontenden med bevidst valgte SSR-, SSG- og CSR-strategier og en genbrugelig factory til prefetchede serversider, der fjerner N+1-kaskaden af fetch-kald fra hver autentificeret side, der flyttes over på den.

Har designet 30 websites fra Photoshop til HTML

Har designet 30 websites og leveret unikke og fleksible løsninger ved at konvertere Photoshop-designs til HTML.

Har udviklet 100 webapplikationer med HTML/HTML5

Har udviklet 100 webapplikationer med HTML/HTML5, CSS/SCSS, Django, WordPress og Joomla og sikret en alsidig onlinetilstedeværelse.

Har bidraget til design af brugergrænseflader

Har bidraget til designprocesser for brugergrænseflader og sikret intuitive, visuelt tiltalende og brugervenlige grænseflader i projekterne.

Har præsenteret 200 UI/UX-forbedringer

Har præsenteret 200 UI/UX-forbedringer til GIS-kortapplikationen og øget omsætningen 10 gange gennem forbedrede softwarefunktioner.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Et afgrænset hændelsesflow i en macOS-app · Har bygget en native macOS-beskedklient i Swift 6 · Har fastlagt platformens grundbeslutninger · Har bygget skift af organisationskontekst · En maritim platform med klare systemgrænser · Har styret full-stack-udviklingen af GIS-kort

{ Softwareudvikling }

Full stack-produktudvikling

Et fungerende produkt fra ende til anden

Ét team, der tager en idé fra skitse til noget, jeres kunder kan logge ind på, i stedet for at kaste den mellem tre leverandører.

Har bygget ejerskabskrav og deres afgørelse

Har bygget ejerskabskrav på virksomheder fra ende til anden — en bruger gør krav på en virksomhed, en administrator afgør sagen, og en godkendelse omskriver den autorisationsgraf, der afgør, hvem der må redigere hvad.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har bygget en native macOS-beskedklient i Swift 6 · Har bygget en databasedrevet CV-generator · Har bygget et Hugo-site på 3 sprog · Har bygget job- og karriereværktøj til søfarende · Har bygget det sociale lag: opslag, feed, grupper · Har bygget platformens operatør-back-office · Har bygget og vedligeholdt Django-apps til IPTV · Har fastlagt platformens grundbeslutninger · Har bygget Next.js 16-frontenden: SSR, SSG og CSR · En maritim platform med klare systemgrænser ·

Et kortprodukt, båret fra ende til anden · Har udviklet 100 webapplikationer med HTML/HTML5 · Har ledet over 30 GIS-projekter · Har styret full-stack-udviklingen af GIS-kort · Har ført en GIS-kortapp til 10x omsætningsvækst

{ Softwareudvikling }

Webstedsudvikling & CMS

Et website, I selv kan rette i

Et site, der er hurtigt, kan findes og kan redigeres af jeres egne folk, uden en regning for hver ændret sætning.

Har udgivet et læsbart Tor-onion-spejl

Har udgivet websitet som et Tor-onion-spejl på en læsbar adresse, der begynder med engineer, med nøglen genereret uden for værten, så den aldrig nåede dette repository eller en laptop, og ladet den være den ene af fire hoveddøre, der aldrig tælles.

Har skabt den visuelle identitet fra 2 tegninger

Har bygget virksomhedens ikon- og favicon-sæt ud fra to håndlavede tegninger, hvor alle afledte filer genskabes af et script, og et tjek fejler, når en afledt fil er committet før sin kilde.

Har bestået 106 af 106 PDF/UA-1-profiltjek

Har dokumenteret PDF/UA-1-validering af ni dokumenter med 106 af 106 bestående kontroller i den konfigurerede tilgængelighedsprofil, mens arkivvarianten fejlede på én af 146 regler.

Har rettet et sitemap med ét fælles tidsstempel

Har rettet et sitemap, hvor 172 af 176 URL'er delte ét ændringstidsstempel, ved at tage datoen fra git-historikken efter at have fastslået, at eksporten omskriver hver fil ved hver kørsel.

Har erstattet 963 metadatablokke med én graf

Har erstattet 963 anonyme blokke med strukturerede data, der gentog virksomheden 1.671 gange, med én sammenkædet graf af 16 typer dannet ud fra stabile oprindelsesidentifikatorer.

Har spejlet sitet til Gemini og Gopher

Har bygget Gemini- og Gopher-output fra samme webkilde; resultatbeskrivelsen angiver 777 dokumenter i hvert format på det pågældende trin og nul ændrede HTML-bytes ved tilføjelsen af formaterne.

Har genereret 495 sider fra en SQLite-eksport

Har genereret 495 resultatsider på tre sprog fra en skrivebeskyttet SQLite-eksport med sidens adresse skrevet som data, så en rettelse i en sætning ikke længere flyttede siden og ødelagde linket.

Jeres kode på jeres egen server

Har udrullet virksomhedens egen git-forge på Soft Serve, privat som udgangspunkt og uden webpanel, med SSH-porten bundet til loopback bag en jump-vært, og gjort landingssiden foran den til et build-artefakt fra hovedsitet i stedet for en manuelt vedligeholdt kopi.

Har kombineret webudvikling med SEO og indhold

Har skabt succes på nettet for kunderne ved at kombinere skræddersyet webudvikling og -design med SEO, indholdsstrategi og tekstforfatning.

Har administreret 40 websites på Apache og Nginx

Har administreret 40 websites på Ubuntu Linux-hostingservere med Apache og Nginx og sikret høj tilgængelighed og ydeevne.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har skåret stylesheet-bundtet fra 87 KB til 31 KB · Har bygget et 637 linjer langt printstylesheet · Har sikret, at reduceret bevægelse virker · De tilgængelighedsfejl, jeres tjekværktøj overser · Tilgængelighedskontroller på 32 sider i begge temaer · Har bygget et Hugo-site på 3 sprog · Har vedligeholdt Hugo og forbedret produktets UI · Har bygget og vedligeholdt Django-apps til IPTV · Har leveret en fuld Progressive Web App · Har bygget Next.js 16-frontenden: SSR, SSG og CSR · Har designet 30 websites fra Photoshop til HTML · Har udviklet 100 webapplikationer med HTML/HTML5

{ Softwareudvikling }

Parathed til AI-agenter

At dukke op, når kunderne spørger en AI

Et site eller en app, som de AI-assistenter, jeres kunder bruger, kan finde, læse og handle på, kontrolleret ved hver udgivelse i stedet for lovet én gang.

Har kørt udvikling som 51 nedskrevne initiativer

Har styret sitets udvikling som 51 nedskrevne initiativer fordelt på 54 plandokumenter, hvert med sine åbne afkrydsningsfelter, sine målinger og de beslutninger, det fravalgte — herunder seks af otte fund om agentparathed, der blev afvist med begrundelsen noteret.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har rettet et sitemap med ét fælles tidsstempel · Har erstattet 963 metadatablokke med én graf · Tilgængelighedskontroller på 32 sider i begge temaer · Har bygget et Hugo-site på 3 sprog

{ Softwareudvikling }

Værktøjer til AI-agenter & MCP-servere

At lade AI bruge den software, I allerede har

Jeres eksisterende software åbnet som værktøjer, en AI-agent kan kalde, uden at noget, der allerede virker, skal skrives om.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har bygget platformens operatør-back-office · Har flyttet langsomme opgaver til en River-jobkø · Har genereret API-kontrakten ud fra databasen · Har migreret API'et til Huma v2 med 100 % paritet · Har designet en JSON-passthrough-arkitektur

{ Softwareudvikling }

GIS & geospatiale løsninger

Kort, der svarer på spørgsmål

Alt, der sker et sted — skibe, køretøjer, matrikler, sensorer — sat på et kort, I kan søge i, måle på og dele.

Har styret 30 succesfulde GIS-projekter

Har styret gennemførelsen af 30 vellykkede GIS-projekter og udvist lederskab ved at levere innovative løsninger inden for fagområdet.

Har optimeret prognose- og investeringsstrategier

Har optimeret prognose- og investeringsstrategier for 11 netselskaber og øget driftseffektiviteten gennem datadrevne GIS-løsninger.

Har automatiseret leveringen af 20 GIS-pipelines

Har automatiseret levering af 20 GIS-datapipelines og ETL-processer til appdata og strømlinet infrastrukturautomatisering og rapportering.

Har automatiseret GIS-SaaS-udrulning med CI/CD

Har automatiseret udrulningen af en GIS SaaS-applikation, databehandlingen og rapporteringssystemet med GitHub Actions CI/CD, Python, Bash og SQL.

Har drevet 100 data warehouse-databaser

Har stået for arkitektur, oprettelse og drift af 100 data warehouse-databaser i PostgreSQL, MS SQL og Google BigQuery, primært med GIS- og tidsseriedata, og optimeret ydeevne og skalerbarhed.

Har udgivet 500 el- og GIS-datarapporter

Har udgivet 500 analyserapporter om el- og GIS-data med grundig research og fejlfinding for at sikre præcis indsigt i store mængder geodata og tidsseriedata.

Har løst 1.000 problemer i geografiske data

Har løst 1.000 problemer i geografiske data og tidsseriedata ved hjælp af GDAL, ArcGIS, PostGIS, Mapbox, QGIS, SQL (PL/pgSQL, Transact-SQL) og Bash, hvilket har sikret behandling af big data i høj kvalitet.

Har forbedret GIS-kortydeevne 10x med PostgreSQL

Har gjort en geografisk kortapplikation 10 gange hurtigere gennem en strategisk databaseovergang fra MSSQL til PostgreSQL, herunder selve omlægningen af data, og optimeret behandlingen og datasikkerheden.

Har gjort en geodatapipeline 50 gange hurtigere

Har registreret en forbedring på 50 gange i en geografisk datapipeline i stifterens arbejde hos Utiligize gennem SQL- og datamodelændringer i PostgreSQL, MS SQL og Google Cloud BigQuery.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Et kortprodukt, båret fra ende til anden · Har ledet over 30 GIS-projekter ·
Har styret full-stack-udviklingen af GIS-kort · Har ført en GIS-kortapp til 10x omsætningsvækst

{ Softwareudvikling }

UI/UX-design & designsystemer

At gøre det indlysende at bruge

Layout, ordvalg og flow sat op, så folk gør færdigt,
hvad de kom efter, i stedet for at ringe til jeres support.

Har renset fejlbeskederne for bebrejdelser

Har omskrevet applikationens fejlbeskeder efter en nedskrevet standard for tone, efter at et afvist login gav brugeren skylden for at have tastet forkert, mens udbyderen i virkeligheden krævede en applikationsspecifik adgangskode, og dækket det med en test, der nævner udbyderen.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har skabt den visuelle identitet fra 2 tegninger · Har bygget et macOS-designsystem med 48 tokens ·
Har navngivet alle 17 ikoner til skærmlæsere · Har bygget en native macOS-beskedklient i Swift 6 ·
Har bestået 106 af 106 PDF/UA-1-profiltjek · Har bygget et 637 linjer langt printstylesheet ·
Har sikret, at reduceret bevægelse virker · Har fundet et manglende lag i det lyse tema ·

Har låst farvesystemet fast på 28 farver · De tilgængelighedsfejl, jeres tjekværktøj overser · Tilgængelighedskontroller på 32 sider i begge temaer · Har erstattet 27 skriftstørrelser med 6 trin · Har kombineret webudvikling med SEO og indhold · Har vedligeholdt Hugo og forbedret produktets UI · Har designet hele produktets UI og UX · Har skabt et iOS 26 'Liquid Glass'-designsystem · Har designet 30 websites fra Photoshop til HTML · Har bidraget til design af brugergrænseflader · Har præsenteret 200 UI/UX-forbedringer

{ Softwareudvikling }

Internationalisering & lokalisering

At virke på mere end ét sprog

Software, der læses naturligt på hvert marked, I sælger på — også datoer, valutaer og den måde, navne sorteres på.

Har bygget et tværsprogligt tjek af tal

Har bygget et tværsprogligt indholdstjek, der fejler, når en oversættelse udelader et tal, som den engelske tekst angiver, og når et sprog bruger en notation, der ikke hører til det — og fundet to danske beskrivelser, der manglede en metrik, og seksten ukrainske tekststykker med citationstegn på engelsk vis.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har flyttet alle tekststrenge til et indholdstræ · Har bygget en databasedrevet CV-generator · Har spejlet sitet til Gemini og Gopher · Har genereret 495 sider fra en SQLite-eksport · Tilgængelighedskontroller på 32 sider i begge temaer · Har bygget et Hugo-site på 3 sprog · Har bygget et engelsk/dansk i18n-system

{ Softwareudvikling }

Udvikling af skræddersyede AI-agenter

En AI-assistent, der kender jeres branche

En agent koblet til de systemer, I bruger, testet på sager, jeres egne eksperter skriver, under opsyn af jeres folk og aldrig låst til én model.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har fået seks indsamlere til at melde det ulæste · Har skrevet tests af selve tjekkene · Har bygget en commit-gate med 22 lintere · Har bygget egen fejlovervågning og tracing · Har bygget en kontrolmotor med 268 tjek

{ Data & databaser }

Databasedesign & datamodellering

Et fornuftigt sted at have sine data

En struktur, der passer til, hvordan forretningen faktisk fungerer, så de svar, I får brug for senere, stadig kan hentes.

Har bygget egen analytics i PostgreSQL

Har bygget egen produktanalyse i PostgreSQL — 47 funktioner over partitionerede event-tabeller, der selv rydder op — pseudonymiseret bag et roterende salt og betinget af den besøgendes samtykke.

Tidsordnede nøgler til en voksende datamodel

Har indført tidsordnede UUID v7-identifikatorer (PostgreSQL 18) som entitetsnøgler for at reducere fragmentering af B-tree-indeks og øge forespørgselshastigheden.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har bygget en databasedrevet CV-generator · Har bygget job- og karriereværktøj til søfarende · Har bygget loyalitetspoint, ligaer og badges · Har flyttet langsomme opgaver til en River-jobkø · Har bygget adgangsret via Stripe og in-app-køb · Har modelleret 348 tabeller i 34 Postgres-skemaer · Har implementeret en katalogdrevet deep-merge ·

Har designet et funktionsbaseret Postgres-datalag · Har designet en JSON-passthrough-arkitektur · En maritim platform med klare systemgrænser · Har drevet 100 data warehouse-databaser · Har designet og drevet 6 ETL/ELT-pipelines

{ Data & databaser }

Databaseadministration (DBA)

At passe på databasen

En ansvarlig for det, hele virksomheden hviler på: den kører, den er sikkerhedskopieret, og den forfalder ikke i det stille.

Har holdt 1.022 migrationer fri for skemadrift

Har holdt skemaet konsistent gennem 1.022 migrationer med en CI-gate, der bygger databasen begge veje — en frisk installation og en installation plus hver eneste migration — og fejler, når de to ikke stemmer overens.

Har implementeret backup og disaster recovery

Har konfigureret opbevaringen af databasebackups som infrastructure-as-code, derefter gennemgået gendannelsesberedskabet og dokumenteret gendannelsesproceduren — og nævnt de resterende huller, i stedet for at de skulle opdages under en hændelse.

Har udviklet 600 PL/pgSQL-ETL/ELT-pipelines

Har udviklet 600 PL/pgSQL-baserede ETL/ELT-pipelines for at strømline komplekse arbejdsgange i databehandlingen i flere udviklings- og produktionsmiljøer med PostgreSQL.

Har automatiseret databehandling i Python og SQL

Har automatiseret databehandlingsopgaver med shell-scripting, PL/pgSQL, Python og Transact-SQL og øget produktiviteten og effektiviteten.

Har automatiseret 100 kritiske databackups

Har automatiseret 100 kritiske databackups ved hjælp af Barman, Google Cloud, Bash og Python og sikret dataintegriteten i alle databaser.

Har gjort PostgreSQL 10 gange hurtigere

Har gjort PostgreSQL 10 gange hurtigere med strategisk indeksering, partitionering og forespørgselsoptimering og effektiviseret databasen til brugerdata, tenantdata, geodata og tidsserier af eldata.

Har skrevet 1.000 RBAC-regler til Linux-servere

Har forbedret datasikkerheden ved at implementere 1.000 RBAC-regler for udviklere, applikationsinstanser, PostgreSQL, MS SQL og andre Linux-servere for at forhindre uautoriseret adgang; dokumenteret med Ansible-automatisering.

Har drevet 10 PostgreSQL- og MS SQL-servere

Har designet, idriftsat og vedligeholdt 10 PostgreSQL- og MS SQL-servere på Ubuntu Linux VPS og sikret optimal ydeevne og driftssikkerhed på serverne.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har drevet 100 data warehouse-databaser

{ Data & databaser }

Databaseoptimering

At gøre de langsomme dele hurtige

At finde de forespørgsler og manglende indeks bag et system, der er blevet tungere hvert år, og rette dem, der betyder noget.

30 dages eldata på 180 ms

Har udviklet varianter til timeaggregering af elforbrug i Python, SQL og Bash + Jq; i benchmarken behandlede Python ét JSONL-datasæt for 30 dage på 180 ms.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Tidsordnede nøgler til en voksende datamodel · Har gjort PostgreSQL 10 gange hurtigere · Har drevet 100 data warehouse-databaser · Har forbedret GIS-kortydeevne 10x med PostgreSQL · Har gjort en geodatapipeline 50 gange hurtigere

{ Data & databaser }

Databasemigrering & modernisering

Væk fra det gamle system

Data taget ud af noget aldrende og ind i noget, der er understøttet, mens forretningen kører videre.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har flyttet alle tekststrenge til et indholdstræ · Har holdt 1.022 migrationer fri for skemadrift · Har forbedret GIS-kortydeevne 10x med PostgreSQL

{ Data & databaser }

Udvikling af datapipelines (ETL/ELT)

Data fra A til B helt af sig selv

Tal, der lander, hvor der er brug for dem, efter en fast plan, uden at nogen kopierer et regneark ved månedens slutning.

Har sparet 4.000 timer med CI/CD i GitHub

Har strømlinet processer for dataanalyse og softwareudvikling og sparet 4.000 timer ved at indføre CI/CD-praksis med GitHub, GitLab, Bash og Python.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har genereret 495 sider fra en SQLite-eksport · Har bygget egen analytics i PostgreSQL · Et kortprodukt, båret fra ende til anden · Har udviklet 600 PL/pgSQL-ETL/ELT-pipelines · Har automatiseret databehandling i Python og SQL · Har ledet over 30 GIS-projekter · Har automatiseret leveringen af 20 GIS-pipelines · Har automatiseret GIS-SaaS-udrulning med CI/CD · Har designet og drevet 6 ETL/ELT-pipelines · Har løst 1.000 problemer i geografiske data · Har gjort en geodatapipeline 50 gange hurtigere · 30 dages eldata på 180 ms · Én infrastrukturplan til 14 afdelinger

{ Data & databaser }

Design af data warehouse

Ét sted, hvor alle tallene stemmer overens

Tal fra salg, drift og økonomi samlet ét sted, så to afdelinger ikke længere kommer til det samme møde med hver sit svar.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har drevet 100 data warehouse-databaser · Har designet og drevet 6 ETL/ELT-pipelines · Har gjort en geodatapipeline 50 gange hurtigere

{ Data & databaser }

Dataanalyse & BI-dashboards

At kunne se, hvad der foregår

Dashboards og rapporter, der viser de få tal, en beslutning faktisk afhænger af, opdateret uden at nogen beder om det.

Har designet et tids- og rapporteringssystem

Har designet et tids- og rapporteringssystem, der forblev i produktion i årevis uden væsentlige ændringer.

Har leveret analyser og løbende statusrapporter

Har leveret analyser og løbende statusrapporter til direktøren og omsat tekniske KPI'er og leveringsstatus til beslutninger.

Har bygget det første observerbarhedsdashboard

Har udviklet og lanceret virksomhedens første dashboard til observerbarhed, der viste systemets ydeevne i realtid og visualiserede data på den store tv-skærm på kontoret.

Brandkommunikation med afsæt i publikums reaktioner

Har brugt markedsundersøgelser og tilbagemeldinger fra publikum til Engineer ApS' brandkommunikation og opsøgende arbejde.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har bygget egen analytics i PostgreSQL · Har udviklet et system til analyserapporter · Har sparet 4.000 timer med CI/CD i GitHub · Har optimeret prognose- og investeringsstrategier · Har udgivet 500 el- og GIS-datarapporter · Har leveret 8 Power BI-projekter med manualer · 30 dages eldata på 180 ms

{ Data & databaser }

Data governance & datakvalitet

At kunne stole på tallene

Regler og kontroller, der fanger dårlige data på vej ind, og et klart svar på, hvem der ejer hvilket tal, og hvor det kommer fra.

Har målt 13 adresser uden tracking

Har målt tretten adresser alene ud fra webserverens egen adgangslog — ingen cookie, ingen tracker og ingen tredjepartstjeneste — hvor klientens netværk maskeres i samme øjeblik linjen skrives, og resultatet samles i et permanent arkiv med 21 daglige tællinger og 20 månedlige facetter.

Kend licensen på alt, hvad I sender ud

Har kontrolleret 644 Rust-crates for licenskompatibilitet ved hvert build og bevist, at tjekket slår til, ved at ændre licensen på én crate og ved at flytte den fastlåste kernerevision uden at regenerere.

Har skrevet tests af selve tjekkene

Har skrevet tests af selve tjekkene efter at have fastslået, at et tjek, der kun fodres med rent input, en dag melder fejlfrit, fordi det intet har læst — ved at plante en stavefejl for at bekræfte, at stavekontrollen finder den, og ved at tage et id-interval fra databasen i stedet for fra et tal i testen.

Har skrevet en afgrænsningsregel efter et læk

Har skrevet en afgrænsningsregel ind i repositoryet, efter at en omstrukturering havde ført en anden virksomheds inventar, firewalltilladelser og tekst med ind i det — og holdt de karantænesatte rester under hemmelighedsscanneren i stedet for at udelukke dem.

Har indsamlet og analyseret forretningskrav

Har indsamlet og analyseret forretningskrav og omsat dem til konkrete funktioner og user stories i overensstemmelse med data governance-standarder.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har bygget et tværsprogligt tjek af tal · Har flyttet alle tekststrenge til et indholdstræ · Har erstattet 963 metadatablokke med én graf · Har bygget egen analytics i PostgreSQL · Har modelleret 348 tabeller i 34 Postgres-skemaer · Har implementeret en katalogdrevet deep-merge · Har designet et funktionsbaseret Postgres-datalag · Har udviklet 600 PL/pgSQL-ETL/ELT-pipelines · Har udgivet 500 el- og GIS-datarapporter · Har løst 1.000 problemer i geografiske data · Én infrastrukturplan til 14 afdelinger

{ Data & databaser }

Klargøring af data til AI

Data, en AI rent faktisk kan bruge

Jeres data samlet, hvor AI-agenter kan nå dem, med én udgave af hver kunde, uden at nogen ser mere, end de må, og med hvert svar sporet tilbage til kilden.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har erstattet 963 metadatablokke med én graf · Har genereret 495 sider fra en SQLite-eksport · Har holdt 1.022 migrationer fri for skemadrift · Har modelleret 348 tabeller i 34 Postgres-skemaer · Har designet og drevet 6 ETL/ELT-pipelines · Har gjort en geodatapipeline 50 gange hurtigere

{ Cloud, infrastruktur & DevOps }

Cloud-infrastruktur & migrering

Der, hvor softwaren kører

Servere dimensioneret efter det, I rent faktisk bruger, flyttet op i skyen eller hjem igen, med den månedlige regning regnet ud først.

Har provisioneret en ekstra server ud fra kode

Har automatiseret provisioneringen af en ekstra server hos en anden cloududbyder og oprettet firewallen før maskinen, så den står bag en firewall fra første øjeblik, med begge udbyderes firewalls skrevet direkte mod deres REST-API'er for at undgå en collection fra tredjepart.

Har kørt containere på Podman og Quadlet

Har udrullet et containerlag på Podman og Quadlet under systemd i stedet for Docker, fordi Dockers standardpublicering af porte på bridge-netværk kan omgå UFW-regler — og givet udrulninger en uprivilegeret bruger med én fast kommando i stedet for root.

Infrastruktur beskrevet i et repository

Har bygget virksomhedens egen infrastruktur som 19 Ansible-playbooks og 34 roller fordelt på 12.065 linjer YAML, der konvergerer den kørende vært til en deklareret tilstand, og hvor hvert play er idempotent.

Har haft ansvaret for alle udrulninger til Azure

Har haft ansvaret for udrulningen af platformen til Azure fra ende til anden og styret releases i udviklings-, staging- og produktionsmiljøet.

Har provisioneret Azure-infrastruktur som kode

Har provisioneret Azure-infrastruktur som kode med Bicep — Container Apps, PostgreSQL Flexible Server, Front Door/WAF og netværk — i udviklings-, staging- og produktionsmiljøet.

Har drevet 20 Docker-apps på Google Cloud og AWS

Har udrullet og vedligeholdt 20 containeriserede Docker-applikationer, fejlsøgt med Podman og Kubernetes og administreret R-baserede apps på Google Cloud og AWS.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Offentlige tjenester på en lille server · Har bygget hele kerneinfrastrukturen fra bunden · Forenklet infrastruktur uden for AWS

{ Cloud, infrastruktur & DevOps }

Infrastructure as Code

Servere, du kan genopbygge ud fra versionsstyret konfiguration

Hele opsætningen skrevet ned som filer, så den kan genskabes præcist og aldrig kun findes i én persons hoved.

Har prøvekørt forgens CI-hook og fundet 2 fejl

Har prøvekørt CI-hooken på forge-siden og fundet to fejl, som ikke kunne findes ved at læse filen: en fallback, der sendte et argument, som ikke kunne opløses, ind i hookens input, og gits egen miljøvariabel, der fulgte gaten ind i checkouten og gjorde 19 tests røde.

Har adskilt alle driftshemmeligheder pr. vært

Har gjort de fire driftshemmeligheder individuelle pr. vært efter at have fastslået, at to værter, der deler én dead man's switch, giver færre alarmer end to switches, ikke flere, og at en fælles adgangssætning til backup kan blotlægge begge værters kopier, selv i separate lagre, hvis nogen får fat i den og kan læse begge lagres krypterede data og nøglefiler.

Har strammet systemd-sandboxen på hver unit

Har reduceret eksponeringen i systemd-sandkassen for hver unit, platformen installerer — en dead man's switch-tjeneste fra 9,6 UNSAFE til 1,5, en git-forge med adgang fra internettet fra 8,3 EXPOSED til 1,5 — og tilføjet et parsertjek ved konvergens efter at have fundet et fejlstavet direktiv, der ubemærket blev ignoreret i tre unit-skabeloner.

Har afstemt en DNS-zone med 20 poster deklarativt

Har afstemt en DNS-zone med 20 poster deklarativt mod Cloudflares API med separate indgange til gennemgang og BIND-eksport og slået CDN-proxyen fra igen af hensyn til privatlivet, efter at den var bygget.

Har tilføjet et preflight-tjek af hemmeligheder

Har tilføjet et preflight-play, der kører den samme kode som konvergensen mod operatørens lokale hemmeligheder på omkring et sekund, efter at en produktionskørsel var gået ned halvvejs, ved sin niende opgave, med swap-indstillingerne allerede skrevet til den kørende vært.

Har fået check-tilstanden til at vise sandheden

Har fået check-tilstanden til at vise sandheden på hele platformen efter at have fundet seks sonder, der traf beslutning ud fra en værdi, værten aldrig havde leveret, fordi Ansibles command-modul melder succes under --check, mens det springer kommandoen helt over.

Har bygget overvågning med dead man's switch

Har bygget overvågning med dead man's switch, der kun pinger, mens hukommelse og disk er i orden, så en svækket vært udløser en alarm ved at gå tavs — og opdaget seks variabelnavne, der sagde "free", hvor tjekket korrekt målte "available", en størrelsesorden til forskel på en maskine med 464 MB.

Backupautomatik med en tydelig gendannelsesmetode

Har implementeret restic-backupautomatik med opbevaringspolitik, integritetskontrol og et forløb til månedlige gendannelsestest og dokumenteret aktivering af lager og kontrol af en virkelig gendannelse som særskilte driftsopgaver.

Har hærdet SSH til 24 validerede direktiver

Har hærdet SSH til 24 direktiver, der alle kontrolleres med assertions, med validering i tre trin — kandidatfilen, den samlede konfiguration og til sidst dæmonens egen tilbagelæsning — efter at tilbagelæsningen afslørede, at den kørende server ubemærket tilsidesatte to af de fireogtyve.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har udgivet et læsbart Tor-onion-spejl · Har provisioneret en ekstra server ud fra kode · Jeres kode på jeres egen server · Offentlige tjenester på en lille server · Infrastruktur beskrevet i et repository · Har implementeret backup og disaster recovery · Har provisioneret Azure-infrastruktur som kode · Har skrevet 1.000 RBAC-regler til Linux-servere

{ Cloud, infrastruktur & DevOps }

DevOps & CI/CD-automatisering

At sende ændringer ud uden drama

Testet, udgivet og til at rulle tilbage, så ændringer når kunderne en tirsdag eftermiddag i stedet for sent en fredag aften.

Har holdt 4 repositories på én historikstandard

Har holdt fire repositories på én standard for historikken — konventionel, uden emojis, uden attributionslinjer, håndhævet af en commit-message-hook — sammen med 48 instruktionsdokumenter, der styrer, hvordan arbejdet udføres.

Har fået testsuiten ned på 8,9 sekunder

Har taget testsuiten fra seks tests over en grænse på tres sekunder til 135 beståede på 8,9 sekunder ved at profilere hovedtråden og fjerne de to kald, den hang i ved 3.989 ud af 4.017 stikprøver.

Har fundet uenighed mellem hooks og kvalitetsgate

Har opdaget, at commit-hooks og kvalitetsgaten kørte forskellige tjek, selvom et dokument lovede, at de var de samme, ved at sammenligne de to lister i en test — de fem, der kun nogensinde blev kørt manuelt, var dem, der læste CV-prosaen.

Har bygget en commit-gate med 22 lintere

Har bygget en commit-gate med 22 lintere, der hver kan beskrives på én linje, plus fem, der fortjener et afsnit, uden advarselsniveau og uden tilladte inline-undertrykkelser, og som dækker HTML, CSS, JavaScript, Python, YAML, Markdown, shell, links, stavning, hemmeligheder og typografi.

Har kørt et løbende sikkerhedsscanningsprogram

Har etableret et løbende sikkerhedsprogram — code scanning, DAST, afhængigheds- og sårbarhedstjek, SBOM-generering, secret scanning og SHA-pinnede actions — sammen med 21 skriftlige sikkerhedsgennemgange.

Har bygget en kontrolmotor med 268 tjek

Har bygget repositoryets kontrolmotor — 268 registrerede commit-tjek, 277 lint-regler og 15 egne ESLint-regler — plus 146 tests af selve tjekkene, så det er builden, der holder standarden, ikke reviewet.

Har automatiseret teamsamarbejde

Har automatiseret teamsamarbejde, adgangskodehåndtering, opgave- og tidsstyring og bygget et halvautomatisk system til fremvisning af projekter og dermed øget teamets produktivitet.

Har indført nul advarsler som krav i seks sprog

Har sat en kvalitetsstandard med nul advarsler i seks sprog — Go, TypeScript, SQL, Python, Shell og Markdown — håndhævet af pre-commit-hooks.

Har skrevet 578 go-task-automatiseringsmål

Har skrevet 578 go-task-automatiseringsmål, der dækker udviklingstilstandene native, Docker og HTTPS, linting, test, database og udrulning.

Har bygget GitHub Actions CI/CD-pipelines

Har bygget GitHub Actions CI/CD-pipelines med et distroless frontend-image til produktion og promovering gennem flere miljøer.

Har strømlinet CI/CD og sparet 4.000 timer

Har strømlinet CI/CD-processer og sparet 4.000 timer ved at indføre automatisering i softwareudviklingspipelines.

Har automatiseret SSL/TLS til 100 Docker-apps

Har automatiseret oprettelsen af SSL/TLS-certifikater til 100 Docker-applikationer og dermed sikret forbindelserne på alle Ubuntu Linux-værter.

Har reduceret driftsrisici med Grafana

Har reduceret driftsrisici ved at implementere et overvågningsdashboard med Grafana og Prometheus og forbedret systemets driftssikkerhed.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Kend licensen på alt, hvad I sender ud · Har taget den ubrugte halvdel af kernen i brug · Har skrevet en parser, der tjekker hvert FFI-kald · Har prøvekørt forgens CI-hook og fundet 2 fejl · Har skrevet tests af selve tjekkene · Har slået alle linterregler til som fejl · Har givet generatoren 981 tests og 92 % dækning · Har lintet selve kvalitetsgaten på 10.242 linjer · Har rettet et sitemap med ét fælles tidsstempel · Har spejlet sitet til Gemini og Gopher · Et sikkerhedsnet, der ikke længere koster en halv time · Har kørt containere på Podman og Quadlet · Jeres kode på jeres egen server · Har tilføjet et preflight-tjek af hemmeligheder · Har fået check-tilstanden til at vise sandheden · Infrastruktur beskrevet i et repository · Har holdt 1.022 migrationer fri for skemadrift · Har bygget egen fejlovervågning og tracing · Har bygget en testsuite til Go, SQL og browseren · Har bygget hele kerneinfrastrukturen fra bunden · Har haft ansvaret for alle udrulninger til Azure · Har genereret API-kontrakten ud fra databasen · Forenklet infrastruktur uden for AWS · Har automatiseret databehandling i Python og SQL · Har sparet 4.000 timer med CI/CD i GitHub · Har fornyet processer og sparet 8.000 timer · Har drevet 20 Docker-apps på Google Cloud og AWS · Har automatiseret 100 kritiske databackups · Har automatiseret leveringen af 20 GIS-pipelines · Har automatiseret GIS-SaaS-udrulning med CI/CD

{ Cloud, infrastruktur & DevOps }

Containerisering & orkestrering

Software, der kører ens alle steder

Pakket, så den opfører sig identisk på en bærbar, en testserver og i produktion, og skalerer op, når der kommer en travl uge.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har kørt containere på Podman og Quadlet · Har strammet systemd-sandboxen på hver unit · Har bygget GitHub Actions CI/CD-pipelines · Har automatiseret SSL/TLS til 100 Docker-apps · Har drevet 20 Docker-apps på Google Cloud og AWS

{ Cloud, infrastruktur & DevOps }

Site reliability & overvågning

At vide det før kunderne gør

Overvågning og alarmer, der siger til, mens fejlen stadig er lille,
og tallene, der viser, hvor tit den bliver ved med at være det.

Har fået seks indsamlere til at melde det ulæste

Har fået seks rapporteringsindsamlere til at oplyse den del af deres input, de aldrig læste — uåbnede filer, ikke-installerede sonder og ugenkendte journallinjer — og tilpasset tærskler til 116 målte stikprøver efter at have fastslået, at en blind indsamling og en rask vært giver den samme side.

Har bygget 7 skrivebeskyttede rapporteringsroller

Har bygget syv skrivebeskyttede rapporteringsroller, der gengiver den kørende vært som Markdown — fakta, adgang, git, metrikker, trafik, sikkerhed og udbyderinventar — efter en regel om, at der ikke udskrives et tal, kørslen ikke selv har målt.

Har bevist, at blokeringen virker ved hver kørsel

Har eftervist blokeringsvejen fra ende til anden ved hver hærdningskørsel ved at blokere en reserveret testadresse, læse den resulterende kerneregel og ophæve blokeringen i en oprydningsblok, der altid kører, så en jail, der holder op med at virke, får kørslen til at fejle i stedet for at melde alt i orden.

Tre låse på døren, ingen af dem låst

Har fundet og rettet tre SSH-beskyttelser mod brute force, der aldrig havde virket: en jail, der holdt øje med port 22, mens dæmonen lyttede på 1986, en ratebegrænsning, som en bredere regel over den overskyggede, og en blokeringshandling, hvis binærfil aldrig blev fundet, så ingen blokering nogensinde var trådt i kraft.

Har sikret uafbrudt levering af IPTV-signaler

Har sikret uafbrudt levering af IPTV-streamingsignaler mellem leverandører og kunder og overvåget og vedligeholdt streamingnetværket og IP-telefonien døgnet rundt.

Har ydet 24/7-support til ca. 1.000 IPTV-servere

Har leveret døgnbemandet 24/7-infrastruktursupport til en IPTV/OTT-streamingplatform og administreret ca. 1.000 servere samt kundeejede systemer for globale kunder i Kina, USA og Tyskland.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har målt 13 adresser uden tracking · Et afgrænset hændelsesflow i en macOS-app ·

Har adskilt alle driftshemmeligheder pr. vært · Har fået check-tilstanden til at vise sandheden ·

Har bygget overvågning med dead man's switch ·

Backupautomatik med en tydelig gendannelsesmetode ·

Har bygget fail-closed rate limiting mod misbrug · Har bygget egen fejlovervågning og tracing ·

Har flyttet langsomme opgaver til en River-jobkø · Har bygget infrastruktur, der holder i årevis ·

Har haft ansvaret for alle udrulninger til Azure · Har bygget e-mail med failover og leveringslog ·

Et kortprodukt, båret fra ende til anden · Har administreret 40 websites på Apache og Nginx · Har reduceret driftsrisici med Grafana · Har drevet 10 PostgreSQL- og MS SQL-servere · Har bygget det første observerbarhedsdashboard

{ Cloud, infrastruktur & DevOps }

Backup & disaster recovery

At få det hele tilbage efter en dårlig dag

Sikkerhedskopier, der er testet og ikke bare antaget at virke, og en nedskrevet plan for den morgen, en server ikke starter igen.

Har administreret 30 Ubuntu Linux VPS-instanser

Har administreret 30 Ubuntu Linux VPS-instanser, implementeret disaster recovery-strategier og sikret optimale netværkskonfigurationer.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har adskilt alle driftshemmeligheder pr. vært · Backupautomatik med en tydelig gendannelsesmetode · Har implementeret backup og disaster recovery · Har automatiseret 100 kritiske databackups

{ Cloud, infrastruktur & DevOps }

Systemadministration

At holde lyset tændt

Daglig drift af servere, mail, konti og opdateringer, så det, alle er afhængige af, forbliver kedeligt.

Har verificeret firewallregler efter position

Har verificeret firewallregler efter position snarere end tilstedeværelse ved at læse den nummererede regelliste og den kørende pakkefilterkæde, fordi det, at en regel findes, ikke betyder, at nogen pakke når frem til den.

Har integreret en fælles adgangskodeløsning

Har integreret et virksomhedsdækkende adgangskodehåndteringssystem, der styrkede sikkerheden og strømlinede adgangskontrollen.

Har administreret 100 barebone-servere

Har administreret 100 barebone-servere, fysiske netværk og IP-telefonisystemer og sikret en robust infrastruktur til virksomhedens vækst.

Har konfigureret og idriftsat 1.000 Wi-Fi-routere

Har konfigureret og idriftsat 1.000 Wi-Fi-routere og forbedret netværkstilgængeligheden og -ydeevnen for kunderne.

Har administreret netværksinfrastruktur

Har administreret netværksinfrastruktur for over 1.000 servere og sikret optimal systemudrulning, sikkerhed og fejlfinding.

Har administreret og fejlsøgt 8 VPN-forbindelser

Har administreret og fejlsøgt 8 WireGuard VPN- og IPsec VPN-forbindelser og sikret beskyttet kommunikation mellem Google Cloud og Linux-systemer.

Har ledet adgangsstyring med M365 og SSO

Har forhindret sikkerhedsbrud ved at lede initiativer inden for adgangsstyring med M365, 1Password, Red Hat SSO og OKTA SSO.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har udgivet et læsbart Tor-onion-spejl · Har fået seks indsamlere til at melde det ulæste · Har prøvekørt forgens CI-hook og fundet 2 fejl · Jeres kode på jeres egen server · Har bygget 7 skrivebeskyttede rapporteringsroller · Har strammet systemd-sandboxen på hver unit · Har afstemt en DNS-zone med 20 poster deklarativt · Har bygget overvågning med dead man's switch · Backupautomatik med en tydelig gendannelsesmetode · Har bevist, at blokeringen virker ved hver kørsel · Har hærdet SSH til 24 validerede direktiver · Offentlige tjenester på en lille server · Infrastruktur beskrevet i et repository · Har bygget hele kerneinfrastrukturen fra bunden · Har bygget infrastruktur, der holder i årevis · Har sikret uafbrudt levering af IPTV-signaler · Har ydet 24/7-support til ca. 1.000 IPTV-servere · Har administreret 40 websites på Apache og Nginx · Har administreret 30 Ubuntu Linux VPS-instanser · Har skrevet 1.000 RBAC-regler til Linux-servere · Har drevet 10 PostgreSQL- og MS SQL-servere

{ Cloud, infrastruktur & DevOps }

Netværk & VPN-opsætning

At forbinde jeres kontorer og jeres folk

Wi-Fi, firewalls og fjernadgang sat op, så medarbejdere kan arbejde hvor som helst, og kun de rette kommer ind.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har provisioneret en ekstra server ud fra kode · Har afstemt en DNS-zone med 20 poster deklarativt · Har verificeret firewallregler efter position · Tre låse på døren, ingen af dem låst · Har sikret uafbrudt levering af IPTV-signaler · Har provisioneret Azure-infrastruktur som kode · Har administreret 100 barebone-servere · Har konfigureret og idriftsat 1.000 Wi-Fi-routere · Har administreret netværksinfrastruktur · Har administreret og fejlsøgt 8 VPN-forbindelser · Har administreret 30 Ubuntu Linux VPS-instanser

{ Cloud, infrastruktur & DevOps }

Sikkerhed & adgangsstyring

At bestemme, hvem der har adgang til hvad

Konti, adgangskoder og rettigheder sat op, så folk har præcis den adgang, jobbet kræver, og fratrådte mister den samme dag.

Har forbedret teamets kommunikation og samarbejde

Har forbedret teamets kommunikation og samarbejde ved at implementere Slack, Mattermost, 1Password og Jira og sparet 8.000 arbejdstimer.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har udgivet et læsbart Tor-onion-spejl · Har målt 13 adresser uden tracking · Kend licensen på alt, hvad I sender ud · Har taget den ubrugte halvdel af kernen i brug · Har adskilt alle driftshemmeligheder pr. vært · Har skrevet en afgrænsningsregel efter et læk · Har provisioneret en ekstra server ud fra kode · Har kørt containere på Podman og Quadlet · Har strammet systemd-sandboxen på hver unit · Har afstemt en DNS-zone med 20 poster deklarativt · Har tilføjet et preflight-tjek af hemmeligheder · Har verificeret firewallregler efter position · Har bevist, at blokeringen virker ved hver kørsel · Har hærdet SSH til 24 validerede direktiver ·

Tre låse på døren, ingen af dem låst · Har kørt et løbende sikkerhedsscanningsprogram · Har bygget ejerskabskrav og deres afgørelse · Har bygget fail-closed rate limiting mod misbrug · Har bygget adgangsret via Stripe og in-app-køb · Har integreret en fælles adgangskodeløsning · Har automatiseret teamsamarbejde · Har hærdet applikationen med nonce-baseret CSP · Har provisioneret Azure-infrastruktur som kode · Har bygget e-mail med failover og leveringslog · Har bygget skift af organisationskontekst · Har automatiseret SSL/TLS til 100 Docker-apps · Har administreret netværksinfrastruktur · Har administreret og fejlsøgt 8 VPN-forbindelser · Har ledet adgangsstyring med M365 og SSO · Har skrevet 1.000 RBAC-regler til Linux-servere

{ Cloud, infrastruktur & DevOps }

Sikkerhed for AI-agenter

At holde AI'en væk fra det, den ikke skal røre

Hver agents adgang kortlagt og begrænset til det nødvendige, hver handling logget, hvor agenten ikke kan ændre loggen, og hver kontrol afprøvet frem for taget for givet.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har adskilt alle driftshemmeligheder pr. vært · Har skrevet en afgrænsningsregel efter et læk · Har tilføjet et preflight-tjek af hemmeligheder · Har bevist, at blokeringen virker ved hver kørsel · Tre låse på døren, ingen af dem låst · Har kørt et løbende sikkerhedsscanningsprogram · Har bygget fail-closed rate limiting mod misbrug · Har ledet adgangsstyring med M365 og SSO · Har skrevet 1.000 RBAC-regler til Linux-servere

{ Rådgivning & ledelse }

Teknisk ledelse & rådgivning

En erfaren ingeniør med i lokalet

En, der sidder på jeres side af bordet, når en leverandør, et budget eller en arkitekturbeslutning kræver en ekstra kvalificeret vurdering.

Har sat dokumentationen under sin egen linter

Har sat dokumentationen under sin egen linter — et linjebudget pr. fil, der kun kan sænkes, et indekskrav og et tjek af faktiske stier — som fandt seks af ni planer markeret som færdige, selvom de stadig havde felter, der ikke var krydset af, og fjorten opgaver, der spærrede hver commit uden at være nævnt i noget dokument.

Har vejledt virksomhedsprojekter i udviklingen

Har vejledt i gennemførelsen af en lang række af virksomhedens projekter og ydet ekspertstøtte i softwareudviklingens faser.

Har ledet væksten fra 4 til 14 medarbejdere

Har ledet virksomhedens vækst fra 4 til 14 medarbejdere ved at anvende agile metoder og effektiv projektledelse.

Har skrevet 50.000 ord softwaredokumentation

Har skrevet 50.000 ord omfattende softwaredokumentation med Markdown i GitHub, Craft og Confluence og derved sikret fastholdelse af viden og gennemsigtighed i processen.

Har anslået 4.000 sparede timer; team 2–18

Har anslået en besparelse på 4.000 timer gennem mentorarbejde i et team, der voksede fra 2 til 18 medlemmer, samt bedre arbejdsgange og koordinering mellem afdelinger.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har holdt 4 repositories på én historikstandard · Har fået testsuiten ned på 8,9 sekunder · Har indført Swift 6 strict concurrency over C · Har fundet uenighed mellem hooks og kvalitetsgate · Har skrevet tests af selve tjekkene · Har slået alle linterregler til som fejl · Har givet generatoren 981 tests og 92 % dækning · Har lintet selve kvalitetsgaten på 10.242 linjer · Har kørt udvikling som 51 nedskrevne initiativer · Et sikkerhedsnet, der ikke længere koster en halv time · Har bygget en kontrolmotor med 268 tjek · Har bygget en testsuite til Go, SQL og browseren · Har leveret analyser og løbende statusrapporter · Har fastlagt platformens grundbeslutninger · Har indført nul advarsler som krav i seks sprog · Har strømlinet CI/CD og sparet 4.000 timer · Har fornyet processer og sparet 8.000 timer · Har ledet over 30 GIS-projekter · Har styret full-stack-udviklingen af GIS-kort · Har ført en GIS-kortapp til 10x omsætningsvækst · Én infrastrukturplan til 14 afdelinger

{ Rådgivning & ledelse }

Teamopbygning & mentoring

At udvikle det team, I allerede har

Ansættelse, oplæring og sparring, så viden bliver hos jeres egne folk i stedet for at forsvinde med en konsulent.

Har ledet et team, der leverede it-support

Har ledet et team, der leverede it-support, datagendannelse og hardwarereparation til over 1.000 kunder og sikret service i høj kvalitet.

Har sparet 150 timer om måneden på 30 projekter

Har forbedret projekteffektiviteten og sparet 150 timer om måneden fordelt på 30 projekter ved at optimere arbejdsgange og ressourcestyring.

Har etableret en teknisk supportafdeling

Har etableret en teknisk supportafdeling, der har betjent over 10.000 kunder med it-support og fejlfinding, og registreret kundehenvendelserne i et CRM-system.

Har designet og styret 5.000 timers kortudvikling

Har designet og styret 5.000 timers kortudvikling ved hjælp af Scrum og Jira for effektiv projektledelse.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har vejledt virksomhedsprojekter i udviklingen ·

Har forbedret teamets kommunikation og samarbejde · Har ledet væksten fra 4 til 14 medarbejdere ·

Har styret 30 succesfulde GIS-projekter · Har anslået 4.000 sparede timer; team 2–18

{ Rådgivning & ledelse }

Projektledelse (Agile)

At få det leveret

En ansvarlig for omfang, datoer og de akavede samtaler, så et projekt har én ejer i stedet for fem meninger.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har holdt 4 repositories på én historikstandard · Har sat dokumentationen under sin egen linter · Har kørt udvikling som 51 nedskrevne initiativer · Har automatiseret teamsamarbejde · Har designet et tids- og rapporteringssystem · Har leveret analyser og løbende statusrapporter · Har sparet 150 timer om måneden på 30 projekter · Har etableret en teknisk supportafdeling · Har fornyet processer og sparet 8.000 timer · Har vejledt virksomhedsprojekter i udviklingen · Har forbedret teamets kommunikation og samarbejde · Har ledet væksten fra 4 til 14 medarbejdere · Har styret 30 succesfulde GIS-projekter · Har indsamlet og analyseret forretningskrav · Har designet og styret 5.000 timers kortudvikling · Har anslået 4.000 sparede timer; team 2–18

{ Rådgivning & ledelse }

Produktstrategi & kravspecifikation

At finde ud af, hvad der skal bygges

At oversætte det, forretningen vil, til noget, en udvikler kan bygge — og sige højt, hvad der ikke er værd at bygge.

En sammenhængende identitet på tværs af kanaler

Har udviklet Engineer ApS' brandidentitet på trykt og digitalt materiale med forbindelse til virksomhedens ingeniørportefølje. Grundlæggeren beskriver en stigning i brandkendskab på 200 gange som resultat af arbejdet.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har renset fejlbeskederne for bebrejdelser · Har kørt udvikling som 51 nedskrevne initiativer · Har bygget en commit-gate med 22 lintere · Har bygget job- og karriereværktøj til søfarende · Har bygget det sociale lag: opslag, feed, grupper · Har bygget loyalitetspoint, ligaer og badges · Har bygget ejerskabskrav og deres afgørelse · Har bygget platformens operatør-back-office · Har bygget adgangsret via Stripe og in-app-køb · Har designet et tids- og rapporteringssystem · Har designet hele produktets UI og UX · Har udviklet et system til analyserapporter · Har indsamlet og analyseret forretningskrav · Har designet og styret 5.000 timers kortudvikling · Har præsenteret 200 UI/UX-forbedringer · Har optimeret prognose- og investeringsstrategier · Har ført en GIS-kortapp til 10x omsætningsvækst · Brandkommunikation med afsæt i publikums reaktioner

{ Rådgivning & ledelse }

Teknisk dokumentation

At skrive det ned, så den næste kan følge med

Overdragelser, driftsvejledninger og manualer, der lader jeres egne folk drive systemet uden at ringe til den, der byggede det.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har fået seks indsamlere til at melde det ulæste · Har holdt 4 repositories på én historikstandard · Har renset fejlbeskederne for bebrejdelser · Har skrevet en parser, der tjekker hvert FFI-kald · Har bygget et tværsprogligt tjek af tal · Har fundet uenighed mellem hooks og kvalitetsgate · Har bestået 106 af 106 PDF/UA-1-profiltjek · Har sat dokumentationen under sin egen linter · Har kørt udvikling som 51 nedskrevne initiativer · Har bygget en commit-gate med 22 lintere · Har skrevet en afgrænsningsregel efter et læk · Har bygget 7 skrivebeskyttede rapporteringsroller · Har kørt et løbende sikkerhedsscanningsprogram · Har bygget en kontrolmotor med 268 tjek · Har skrevet 578 go-task-automatiseringsmål · Har migreret API'et til Huma v2 med 100 % paritet · Har skrevet 50.000 ord softwaredokumentation · Har leveret 8 Power BI-projekter med manualer · Én infrastrukturplan til 14 afdelinger

{ It-support & hardware }

It-support & helpdesk

Almindelig computerhjælp

Den bærbare, printeren, login og postkassen klaret hurtigt og i et sprog, man forstår, uden at nogen føler sig dum.

Har diagnosticeret og repareret bærbare computere

Har diagnosticeret og repareret hardware i bærbare computere — skærme, hængsler, tastaturer, trackpads, bundkort og strømforsyninger — og løst softwareproblemer i Linux, Windows og Mac.

Har gendannet data fra harddiske, SSD'er og RAID

Har udført datagendannelse på en bred vifte af medier — SD-kort, harddiske, SSD'er, RAID-arrays, eksterne drev og Mac-systemer.

Har håndteret 4.000 computerreparationer

Har håndteret 4.000 computerreparationer og sikret hurtig og effektiv løsning af hardware- og softwareproblemer.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har ydet 24/7-support til ca. 1.000 IPTV-servere · Har konfigureret og idriftsat 1.000 Wi-Fi-routere · Har ledet et team, der leverede it-support · Har etableret en teknisk supportafdeling

{ It-support & hardware }

Hardwarereparation & datagendannelse

At redde filer ud af en død maskine

Reparation af det, der kan betale sig at reparere, og at få billeder, fakturaer eller tegninger ud af en disk, der er gået i stå.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har diagnosticeret og repareret bærbare computere · Har gendannet data fra harddiske, SSD'er og RAID · Har håndteret 4.000 computerreparationer · Har ledet et team, der leverede it-support

{ Brand & marketing }

Brand, marketing & SEO

At blive fundet og se rigtig ud

Et navn, et udtryk og et website, der dukker op, når nogen søger efter det, I sælger, og læses ens alle de steder, I optræder.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har skabt den visuelle identitet fra 2 tegninger · Har bygget et macOS-designsystem med 48 tokens · Har rettet et sitemap med ét fælles tidsstempel · Har skåret stylesheet-bundtet fra 87 KB til 31 KB · Har erstattet 963 metadatablokke med én graf · Har låst farvesystemet fast på 28 farver · Har genereret 495 sider fra en SQLite-eksport · Har erstattet 27 skriftstørrelser med 6 trin ·

Har kombineret webudvikling med SEO og indhold ·

Brandkommunikation med afsæt i publikums reaktioner ·

En sammenhængende identitet på tværs af kanaler

Sammenlign jeres udgangspunkt

Vælg et sammenligneligt arbejde, og identificér forskellene i jeres egen situation.

Se, hvad det koster Kontakt os →

Drøft en relevant tilgang

Bed os omsætte relevant erfaring til en tilgang, I kan vurdere.

Se vores ydelser

welcome@engineer.company +45 50 10 61 84 København, Danmark

Tag denne side med dig

Visitkort

Lavet med kærlighed af **D.**

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

Copyright © 2025 – nu · Engineer ApS · Alle rettigheder forbeholdes.

Udskrevet fra https://engineer.company/da/portfolio/