Engineer Data

Portfölj

Ingenjörstjänster med exempel på utfört arbete.

Engineer ApS registrerades 2025. Merparten av arbetet som visas här levererades dessförinnan av vår grundare inne i kundernas egna team; en del har levererats av företaget självt, och en del sedan dess i anställning. Samma händer och samma krav — vi säger vad som är vad i stället för att sudda ut det.

Hitta erfarenhet värd att använda

Använd översikten för att hitta erfarenhet som är relevant för ert projekt.

Se våra tjänster Hör av er →

Läs besluten bakom arbetet

Exemplen beskriver levererat arbete över flera tjänster. Läs en genomgång för att se situationen och besluten.

Fallstudier Hör av er →

Bläddra efter kategori

MJUKVARUUTVECKLING

Backendutveckling · Frontendutveckling · Fullstackutveckling · API:er och integration · Webbutveckling · Test och kvalitetssäkring · Prestandaoptimering · Python · Internationalisering

DATA & DATABASER

Databaser · PostgreSQL · SQL · Data engineering · Datapipelines (ETL/ELT) · Dataanalys · Datastyrning · Datalager · GIS / geodata

MOLN, INFRASTRUKTUR & DEVOPS

Molntjänster · Containrar (Docker/Kubernetes) · DevOps · Automatisering och CI/CD · Infrastruktur · Linux och servrar · Systemadministration · Nätverk och VPN · Övervakning och observerbarhet · Tillförlitlighet och säkerhetskopiering · Säkerhet · Plattformsarkitektur · Migrering och modernisering

RÅDGIVNING & LEDARSKAP

Tekniskt ledarskap · Teamledning · Agilt arbete och Scrum · Projektledning · Produkt och krav · Intressenter och rapportering · Dokumentation · Lösningsarkitektur · Varumärke och marknadsföring

Allt nedan **är levererat** — inte föreslaget, inte planerat.

Varje post berättar vad arbetet förändrade och länkar till tjänsterna den visar.

Det vi erbjuder Vad det kostar →

{ Data & databaser }

Databasdesign och datamodellering

Ett vettigt ställe att förvara dina data

En struktur som speglar hur verksamheten faktiskt fungerar, så att de frågor du behöver svar på senare fortfarande går att ställa.

Byggde en databasdriven CV-generator

Byggde en databasdriven generator för CV, referenser, portfolio och personliga brev i Python — 41 moduler, 10 580 rader — som renderar sex utdataformat från en SQLite-källa med 23 tabeller, sammanställd av en idempotent pipeline i 19 steg.

Byggde rekryterings- och karriärytor för sjöfolk

Byggde arbetsytor för rekrytering och sjöfolks karriär; den dokumenterade implementationsbilden omfattar 151 lagrade funktioner i 48 tabeller för tjänster, ansökningar, certifikat, rangutveckling och sjötid.

Byggde en lojalitetsplånbok, ligor och märken

Byggde lojalitets- och ryktessystemet — 67 funktioner över en huvudbok i 31 tabeller, med ligor, märken och en inlösenbutik — som tar ett radlås på saldot för att stänga fönstret för dubbelspendering.

Byggde egen analys inuti PostgreSQL

Byggde egen produktanalys i PostgreSQL — 47 funktioner över partitionerade händelsetabeller som rensar sig själva — pseudonymiserad bakom ett roterande salt och villkorad av besökarens samtycke.

Flyttade långsamt arbete till en jobbkö i River

Flyttade långsamt arbete från anropsvägen till en jobbkö i River — 15 workermoduler, 8 schemalagda uppgifter och 20 pg_cron-jobb — så att ett anrop returnerar medan arbetet bakom det fortsätter.

Byggde åtkomsträttigheter via Stripe och appköp

Byggde skiktet för betalningar och åtkomsträttigheter — Stripe vid sidan av köp i appen via Apple och Google — som spärrar katalogen, sökningen och exporten bakom en åtkomstmodell i 11 tabeller som kontrolleras på servern.

Modellerade 348 tabeller i 34 Postgres-scheman

Modellerade sjöfartsdomänen i 348 normaliserade tabeller i 34 PostgreSQL-scheman — yrkesverksamma, företag, fartyg, jobb, recensioner och resten — med SMALLINT-uppslag och UUID v7-nycklar.

Implementerade en katalogstyrd djupsammanslagning

Implementerade en katalogstyrd djupsammanslagning av lagrade JSON-inställningar som förhindrar krascher på grund av saknade nycklar när schemat utvecklas.

Tidsordnade nycklar för en växande datamodell

Införde tidsordnade UUID v7-identifierare (PostgreSQL 18) som entitetsnycklar för att minska fragmenteringen i B-trädindex och snabba upp databasfrågor.

Utformade ett funktionsbaserat skikt i PostgreSQL

Utformade ett funktionsbaserat dataskikt i PostgreSQL — 1 275 lagrade funktioner i 34 scheman — så att varje läsning och skrivning går via en funktion som databasen kan ge behörighet till, i stället för via en tabell.

Utformade en JSON-väg rakt från Postgres till API

Utformade en arkitektur för direkt vidarebefordran av JSON, där PostgreSQL-funktioner returnerar komplett JSON som Go-API:t skickar vidare ordagrant, vilket tog bort mellanliggande deserialisering och frikopplade frontenden från schemaändringar.

En maritim plattform med tydliga systemgränser

Utformade arkitekturen för en skiktad sjöfartsplattform som håller isär en PWA-frontend i Next.js, ett API i Go (Huma/Fiber) och ett funktionsskikt i PostgreSQL, med all affärslogik i databasen.

Drev 100 datalager i PostgreSQL, MS SQL, BigQuery

Utformade, byggde upp och förvaltade 100 datalagerdatabaser i PostgreSQL, MS SQL och Google BigQuery, främst med GIS- och tidsseriedata, och optimerade prestanda och skalbarhet.

Utformade och drev 6 ETL/ELT-pipelines

Utformade, implementerade och administrerade 6 ETL/ELT-pipelines med Google BigQuery, MSSQL, PostgreSQL, Shell-skript, PL/pgSQL och Transact-SQL och integrerade data för effektiv bearbetning i ett Python-API.

{ Data & databaser }

Databasadministration (DBA)

Någon som tar hand om databasen

Någon som ansvarar för det som hela företaget vilar på: den är i drift, den är säkerhetskopierad och den förfaller inte i det tysta.

1 022 migreringar bakom en grind mot schemadrift

Höll schemat korrekt genom 1 022 migreringar med en CI-grind som bygger databasen på båda sätten — en ny installation, och en installation plus varje migrering — och fäller när de två skiljer sig åt.

Byggde säkerhetskopior och återställning som kod

Konfigurerade lagringstiden för databasens säkerhetskopior som infrastruktur som kod, granskade sedan återställningsförmågan och dokumenterade rutinen för återläsning — och pekade ut de återstående luckorna i stället för att låta dem upptäckas under en incident.

Byggde 600 ETL/ELT-pipelines i PL/pgSQL

Byggde 600 PL/pgSQL-baserade ETL/ELT-pipelines för att effektivisera komplexa arbetsflöden för databearbetning i flera utvecklings- och produktionsmiljöer för PostgreSQL.

Automatiserade databearbetning i Python, PL/pgSQL

Automatiserade uppgifter inom databearbetning med Shell-skript, PL/pgSQL, Python och Transact-SQL och ökade produktiviteten och effektiviteten.

Automatiserade 100 kritiska säkerhetskopior

Automatiserade 100 kritiska säkerhetskopior av data med Barman, Google Cloud, Bash och Python och säkerställde dataintegriteten i samtliga databaser.

Gjorde PostgreSQL 10 gånger snabbare

Gjorde PostgreSQL 10 gånger snabbare med strategisk indexering, partitionering och optimering av databasfrågor och effektiviserade databasen för användar- och tenantdata, geospatiala data och tidsseriedata om el.

Skrev 1 000 RBAC-regler som säkrar Linux-servrar

Stärkte datasäkerheten genom att införa 1 000 RBAC-regler för utvecklare, applikationsinstanser, PostgreSQL, MS SQL och andra Linux-servrar, vilket förhindrade obehörig åtkomst; dokumenterade reglerna som automatisering i Ansible.

Drev 10 PostgreSQL- och MS SQL-servrar på Ubuntu

Utformade, driftsatte och underhöll 10 PostgreSQL- och MS SQL-servrar på VPS:er med Ubuntu Linux och säkerställde optimal serverprestanda och tillförlitlighet.

Tjänsten ingår i arbete som står annanstans på sidan:

Drev 100 datalager i PostgreSQL, MS SQL, BigQuery

{ Data & databaser }

Optimering av databasprestanda

Gör de långsamma delarna snabba

Hittar databasfrågorna och de saknade indexen bakom ett system som har blivit tyngre för varje år, och åtgärdar dem som spelar roll.

Gjorde GIS-kartan 10 gånger snabbare i PostgreSQL

Förbättrade prestandan i en kartapplikation för geodata 10 gånger genom en strategisk databasövergång från MSSQL till PostgreSQL, inklusive själva omställningen av data, och optimerade bearbetningen och datasäkerheten.

Gjorde pipelinen för geodata 50 gånger snabbare

Dokumenterade en förbättring på 50 gånger i en geografisk datapipeline i grundarens arbete hos Utiligize genom SQL- och datamodelländringar i PostgreSQL, MS SQL och Google Cloud BigQuery.

30 dagars eldata på 180 ms

Utvecklade varianter för timvis aggregering av elförbrukning i Python, SQL och Bash + Jq; Python-varianten behandlade en JSONL-datamängd för 30 dagar på 180 ms i prestandatestet.

Tjänsten ingår i arbete som står annanstans på sidan:

Tidsordnade nycklar för en växande datamodell · Gjorde PostgreSQL 10 gånger snabbare · Drev 100 datalager i PostgreSQL, MS SQL, BigQuery

{ Data & databaser }

Databasmigrering och modernisering

Bort från det gamla systemet

Data som flyttas ur något åldrande och in i något som fortfarande underhålls, medan verksamheten fortsätter som vanligt under tiden.

Flyttade varje sträng till ett innehållsträd

Flyttade varje sträng som användarna ser ut ur Python till ett innehållsträd med 874 filer på tre språk, efter att ha hittat döda översättningar som ingen kunde se var döda och en kontroll som i tysthet bedömde en tredjedel av resultaten.

Tjänsten ingår i arbete som står annanstans på sidan:

1 022 migreringar bakom en grind mot schemadrift ·

Gjorde GIS-kartan 10 gånger snabbare i PostgreSQL

{ Data & databaser }

Utveckling av datapipelines (ETL/ELT)

Data som tar sig från A till B av sig själv

Siffror som kommer fram dit de behövs, på fasta tider, utan att någon kopierar ett kalkylblad i slutet av månaden.

Genererade 495 sidor från en SQLite-export

Genererade 495 resultatsidor på tre språk från en skrivskyddad SQLite-export, med sidadressen angiven som data så att en rättad mening inte längre flyttade sidan och bröt länken.

En kartprodukt, buren hela vägen

Utformade, utvecklade och stödde infrastruktur, databearbetning och en kartapplikation under 2 år, med både programvaruleveranser och driftunderhåll.

Sparade 4 000 timmar med GitHub och GitLab CI/CD

Effektiviserade processerna för dataanalys och mjukvaruutveckling och sparade 4 000 timmar genom att införa CI/CD-arbetssätt med GitHub, GitLab, Bash och Python.

Ledde över 30 GIS-projekt från bygge till support

Ledde utveckling, driftsättning och support av över 30 GIS-projekt och visade expertis inom PostgreSQL, Bash, Python, JavaScript, GDAL, ArcGIS, PostGIS och Mapbox.

Automatiserade leveransen av 20 GIS-datapipelines

Automatiserade leveransen av 20 GIS-datapipelines och ETL-processer för applikationsdata, vilket effektiviserade infrastrukturautomatiseringen och rapporteringen.

Automatiserade GIS-driftsättning i GitHub Actions

Automatiserade driftsättningen av en GIS-applikation som SaaS, databearbetningen och rapporteringssystemet med GitHub Actions CI/CD, Python, Bash och SQL.

Löste 1 000 problem i geodata och tidsseriedata

Löste 1 000 problem i geodata och tidsseriedata med GDAL, ArcGIS, PostGIS, Mapbox, QGIS, SQL (PL/pgSQL, Transact-SQL) och Bash, vilket säkrade en högkvalitativ bearbetning av stora datamängder.

En infrastrukturplan för 14 avdelningar

Utformade ett heltäckande infrastrukturramverk för ett forskningsinstitut som berörde 14 avdelningar, med flexibla moduler, enhetliga datapipelines och strukturerade supportstrategier för ett långsiktigt införande.

Tjänsten ingår i arbete som står annanstans på sidan:

Byggde egen analys inuti PostgreSQL · Byggde 600 ETL/ELT-pipelines i PL/pgSQL · Automatiserade databearbetning i Python, PL/pgSQL · Utformade och drev 6 ETL/ELT-pipelines · Gjorde pipelinen för geodata 50 gånger snabbare · 30 dagars eldata på 180 ms

{ Data & databaser }

Design av datalager

Ett ställe där alla siffror stämmer överens

Siffror från försäljning, drift och ekonomi samlade på ett ställe, så att två avdelningar slutar komma med två olika svar till samma möte.

Tjänsten ingår i arbete som står annanstans på sidan:

Drev 100 datalager i PostgreSQL, MS SQL, BigQuery · Utformade och drev 6 ETL/ELT-pipelines · Gjorde pipelinen för geodata 50 gånger snabbare

{ Data & databaser }

Dataanalys och BI-dashboards

Se vad som händer

Dashboards och rapporter som visar de få siffror ett beslut faktiskt hänger på, uppdaterade utan att någon behöver be om det.

Utformade ett tids- och rapporteringssystem

Utformade ett system för tidshantering och rapportering som användes i produktion i flera år utan större ändringar.

Levererade analyser och lägesrapporter till vd:n

Levererade analyser och regelbundna lägesrapporter till vd:n och omsatte tekniska nyckeltal och leveransstatus i beslut.

Utvecklade ett rapportsystem för dataanalys

Utvecklade ett rapportsystem för dataanalys som ökade de kvartalsvisa mjukvaruintäkterna med 400 % genom Python-baserade PDF-rapporter.

Optimerade prognos- och investeringsstrategier

Optimerade prognos- och investeringsstrategier för 11 elnätsföretag och drev på driftseffektiviteten med datadrivna GIS-lösningar.

Publicerade 500 rapporter om el- och GIS-data

Publicerade 500 analysrapporter om el- och GIS-data och använde grundlig research och felsökning för att säkerställa korrekta insikter ur stora mängder geodata och tidsseriedata.

Levererade 8 Power BI-projekt med handböcker

Levererade 8 Power BI-projekt med utförliga handböcker och integrerade Microsoft Power BI-verktygen med ett NodeJS-API och Python FastAPI för effektiv dataanalys och visualisering.

Byggde företagets första observerbarhetsdashboard

Utvecklade och lanserade företagets första dashboard för observerbarhet, med realtidsinsikt i systemens prestanda och datavisualisering på den stora tv:n på kontoret.

Varumärkeskommunikation utifrån publikens respons

Använde marknadsundersökningar och publikens återkoppling i Engineer ApS varumärkeskommunikation och uppsökande arbete.

Tjänsten ingår i arbete som står annanstans på sidan:

Byggde egen analys inuti PostgreSQL · Sparade 4 000 timmar med GitHub och GitLab CI/CD · 30 dagars eldata på 180 ms

{ Data & databaser }

Datastyrning och datakvalitet

Kunna lita på siffrorna

Regler och kontroller som fångar dålig data på vägen in, och ett tydligt svar på vem som äger vilken siffra och var den kommer ifrån.

Mätte 13 adresser helt utan spårning

Mätte tretton adresser enbart utifrån webbserverns egen åtkomstlogg — ingen cookie, ingen spårare och ingen tjänst från tredje part — maskerade klientens nätverk när varje rad skrivs och förde in resultatet i ett permanent arkiv med 21 dagliga räkningar och 20 månatliga fasetter.

Känn till licensen för allt ni levererar

Granskade licenskompatibiliteten hos 644 Rust-crates vid varje bygge och bevisade att kontrollen slår till genom att skriva om licensen för en crate och genom att flytta den låsta kärnrevisionen utan att generera om.

Byggde en sifferkontroll mellan språken

Byggde en innehållskontroll mellan språken som fäller när en översättning tappar en siffra som den engelska texten anger, och när ett språk använder en notation som det inte använder — och hittade två danska beskrivningar som saknade ett mätvärde och sexton ukrainska textavsnitt som citerade på engelskt sätt.

Skrev tester för själva kontrollerna

Skrev tester för själva kontrollerna efter att ha konstaterat att en kontroll som bara matas med felfri indata en dag rapporterar felfritt för att den inte läste något — lade in en avsiktlig felstavning för att bekräfta att stavningskontrollen hittar den och hämtade ett id-intervall från databasen i stället för från ett tal i testet.

Ersatte 963 metadatablock med en enda graf

Ersatte 963 anonyma block av strukturerad data som upprepade företaget 1 671 gånger med en enda länkad graf av 16 typer, präglade från stabila ursprungsidentifierare.

Skrev en avgränsningsregel efter läckta uppgifter

Skrev in en avgränsningsregel i repot efter att en omstrukturering hade fört in ett annat företags värdförteckning, brandväggsundantag och text i det — och höll de karantänsatta resterna kvar under skanningen efter hemligheter i stället för att undanta dem.

Samlade in och analyserade verksamhetskrav

Samlade in och analyserade verksamhetskrav och omsatte dem i genomförbara funktioner och user stories i linje med standarderna för datastyrning.

Tjänsten ingår i arbete som står annanstans på sidan:

Flyttade varje sträng till ett innehållsträd · Byggde egen analys inuti PostgreSQL ·

Modellerade 348 tabeller i 34 Postgres-scheman ·

Implementerade en katalogstyrd djupsammanslagning ·

Utformade ett funktionsbaserat skikt i PostgreSQL · Byggde 600 ETL/ELT-pipelines i PL/pgSQL · Publicerade 500 rapporter om el- och GIS-data · Löste 1 000 problem i geodata och tidsseriedata · En infrastrukturplan för 14 avdelningar

{ Data & databaser }

Dataförberedelse för AI

Data som en AI faktiskt kan använda

Dina data samlade där AI-agenter kan nå dem, med en enda version av varje kund, utan att någon ser mer än de får, och med varje svar spårat tillbaka till källan.

Tjänsten ingår i arbete som står annanstans på sidan:

Ersatte 963 metadatablock med en enda graf · Genererade 495 sidor från en SQLite-export · 1 022 migreringar bakom en grind mot schemadrift · Modellerade 348 tabeller i 34 Postgres-scheman · Utformade och drev 6 ETL/ELT-pipelines · Gjorde pipelinen för geodata 50 gånger snabbare

Jämför ert utgångsläge

Välj ett jämförbart arbete och identifiera skillnaderna i er egen situation.

Se vad det kostar Hör av er →

Diskutera ett relevant angreppssätt

Be oss göra relevant erfarenhet till ett angreppssätt ni kan bedöma.

Se våra tjänster

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

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://data.engineer.company/sv/portfolio/