Engineer Data

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.

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 →

Se efter kategori

SOFTWAREUDVIKLING

Backend-udvikling · Frontend-udvikling · Full stack-udvikling · API'er & integration · Webudvikling · 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 · 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 →

{ 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 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.

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 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 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.

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 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 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.

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.

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 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.

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 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.

{ 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.

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.

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

{ 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.

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.

Ydelsen indgår i arbejde, der står andetsteds på siden:

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 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.

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 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.

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 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 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.

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.

Ydelsen indgår i arbejde, der står andetsteds på siden:

Har bygget egen analytics i PostgreSQL · Har udviklet 600 PL/pgSQL-ETL/ELT-pipelines · Har automatiseret databehandling i Python og SQL · Har designet og drevet 6 ETL/ELT-pipelines · Har gjort en geodatapipeline 50 gange hurtigere · 30 dages eldata på 180 ms

{ 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 udviklet et system til analyserapporter

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

Har optimeret prognose- og investeringsstrategier

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

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 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 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 sparet 4.000 timer med CI/CD i GitHub ·

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 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.

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 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 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 flyttet alle tekststrenge til et indholdstræ · 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

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://data.engineer.company/da/portfolio/