Engineer Advisory

Portfolio

Engineering services, with examples of delivered work.

Engineer ApS was registered in 2025. Most of the work shown here was delivered before that by our founder inside client teams; some has been delivered by the company itself, and some since in employment. The same hands and the same standards — we say which is which rather than blurring it.

Find experience worth applying

Use this record to find experience that is relevant to your project.

Explore our services Get in touch →

Read the decisions behind the record

The entries describe delivered work across services. Read a review to see the situation and decisions behind it.

Cases Get in touch →

Browse by category

SOFTWARE ENGINEERING

Backend Engineering
Frontend Engineering
Full-Stack Development
APIs & Integration
Web Development
Design Systems & UI
UX / UI Design
Testing & QA
Performance Tuning
Python
Internationalization

DATA & DATABASES

Databases
PostgreSQL
SQL
Data Engineering
Data Pipelines (ETL/ELT)
Data Analytics
Data Governance
GIS / Geospatial

CLOUD, INFRASTRUCTURE & DEVOPS

DevOps · Automation & CI/CD · Infrastructure · Linux & Servers · System Administration · Monitoring & Observability · Reliability & Backups · Security · Platform Architecture · Migrations & Modernization

CONSULTING & LEADERSHIP

Technical Leadership · Team Leadership · Mentoring & Coaching · Agile & Scrum · Project Management · Product & Requirements · Stakeholder & Reporting · Documentation · Solution Architecture · Brand & Marketing

Everything below **was delivered** — not proposed, not planned.

Each entry names what the work changed and links the services it demonstrates.

What we offer What it costs →

{ Consulting & Leadership }

Technical Leadership & Consulting

A senior engineer in the room

Somebody on your side of the table when a supplier, a budget or an architecture decision needs a second informed opinion.

Held 4 repositories to one history standard

Held four repositories to one history standard — conventional, emoji-free, no attribution trailers, enforced by a commit-message hook — alongside 48 instruction documents that govern how the work is done.

Took the test suite to 8.9 seconds

Took the test suite from six tests over a sixty-second limit to 135 passing in 8.9 seconds by profiling the main thread and removing the two calls it sat inside for 3,989 samples out of 4,017.

Adopted Swift 6 strict concurrency over C

Adopted Swift 6 complete strict concurrency with no actors, bridging a blocking C event loop to the main actor through one producer, one consumer and one ordering — after establishing that a task per event loses the ordering the interface depends on.

Found the hooks and the gate disagreeing

Found the commit hooks and the quality gate running different checks while a document promised they were the same, by comparing the two lists in a test — the five that only ever ran by hand were the ones reading the CV prose.

Wrote tests for the checkers themselves

Wrote tests for the checkers themselves after establishing that a checker fed only clean input will one day report clean because it read nothing — planting a misspelling to confirm the spell-check finds it, and taking an id range from the database rather than from a number in the test.

Enabled every linter rule as an error

Selected every rule the Python linter has as an error, working through 1,815 findings to reach zero, with each of the few exemptions carrying a written reason and two of them backed by a checker instead of a comment.

Held the generator to 981 tests and 92% coverage

Held the generator to 981 test cases at a 92% branch-coverage floor with warnings treated as failures, and asserted idempotence by running the whole build pipeline twice from an empty file and requiring the second pass to change nothing.

Linted the 10,242-line quality gate itself

Brought 10,242 lines of quality-gate JavaScript under a formatter and a linter after establishing it was the largest body of code in the repository and the only one nothing read, fixing 13 findings and suppressing none.

Put the documentation under its own linter

Put the documentation under its own linter — a per-file line budget that only ratchets down, an index requirement and a real-path check — which found six of nine plans marked done while carrying unchecked boxes, and fourteen commit-gating tasks named in no document.

Ran development as 51 written initiatives

Ran the site's development as 51 written initiatives across 54 plan documents, each carrying its open boxes, its measurements and the decisions it declined — including six of eight agent-readiness findings refused with the reason recorded.

A safety net that no longer costs half an hour

Cut the site's browser-driven quality gate from 1,636 seconds to 615 by scheduling its checks longest-first through a worker pool bounded to four lanes, after measuring that alphabetical order cost 320 seconds against 224.

Built a 268-check guard engine for the repo

Built the repository's guard engine — 268 registered commit checks, 277 lint rules and 15 custom ESLint rules — plus 146 tests of the guards themselves, so the build holds the standard, not review.

Built a test suite across Go, SQL and browser

Built a layered automated test suite — 981 Go tests, 543 frontend and browser specs, 494 SQL behavioural tests — with mutation testing, property-based tests and an accessibility gate.

Delivered analysis and regular progress reports

Delivered analysis and regular progress reports to the CEO, translating engineering KPIs and delivery status into decisions.

Set the platform's founding technical decisions

Set the platform's founding decisions in the weeks after the repository opened in November 2025 — the layering, database-first data access and the zero-warnings bar — and they still hold nine months on.

Set a zero-warnings bar across six languages

Set a zero-warnings quality bar across six languages — Go, TypeScript, SQL, Python, Shell and Markdown — enforced by pre-commit hooks.

Streamlined CI/CD processes, saving 4,000 hours

Streamlined CI/CD processes, saving 4,000 hours by introducing automation in software development pipelines.

Overhauled internal processes, saving 8,000 hours

Overhauled internal processes, saving 8,000 hours by improving software architecture, systems, and scheduling efficiency.

Guided company projects through development

Guided the execution of numerous company projects, offering expert support in the software development phases.

Led 30+ GIS projects from build to support

Led the development, deployment, and support of over 30 GIS projects, demonstrating expertise in PostgreSQL, Bash, Python, JavaScript, GDAL, ArcGIS, PostGIS, and Mapbox technologies.

Led company growth from 4 to 14 employees

Led company growth from 4 to 14 employees by employing Agile methodologies and effective project management practices.

Wrote 50,000 words of software documentation

Wrote 50,000 words of comprehensive software documentation utilizing Markdown in GitHub, Craft, and Confluence, thereby ensuring the retention of knowledge and the transparency of the process.

Estimated 4,000 hours saved, team from 2 to 18

Estimated 4,000 hours saved while mentoring a team that grew from 2 to 18 members, improving workflows and cross-departmental coordination.

Directed full-stack GIS map development

Directed full-stack GIS map development, overseeing PostgreSQL, Mapbox, ReactJS, and NodeJS to deliver an integrated solution.

Led a GIS map application to 10x revenue growth

Led the software development of a GIS map application, driving revenue growth by 10x and positioning the product as a primary data asset.

One infrastructure plan for 14 departments

Designed a comprehensive infrastructure framework for a research institute impacting 14 departments, featuring flexible modules, unified data pipelines, and structured support strategies for long-term adoption.

{ Consulting & Leadership }

Team Building & Mentoring

Growing the team you already have

Hiring, onboarding and coaching, so the knowledge settles with your own people rather than leaving with a consultant.

Managed a team delivering IT support

Managed a team delivering IT support, data recovery, and hardware repair services to over 1,000 clients, ensuring high-quality service.

Saved 150 hours a month across 30 projects

Enhanced project efficiency, saving 150 hours per month across 30 projects by optimizing workflows and resource management.

Established a Technical Support department

Established a Technical Support department, servicing over 10,000 clients with IT support and troubleshooting solutions, tracking client requests in a CRM.

Improved team communication and collaboration

Improved team communication and collaboration by implementing Slack, Mattermost, 1Password, and Jira, saving 8,000 hours of labor.

Managed 30 successful GIS projects to delivery

Managed the execution of 30 successful GIS projects, demonstrating leadership in delivering innovative solutions across the field.

Managed 5,000 hours of map development

Designed and managed 5,000 hours of map development, using Scrum and Jira for effective project management.

This service is part of work listed elsewhere on this page:

Guided company projects through development · Led company growth from 4 to 14 employees · Estimated 4,000 hours saved, team from 2 to 18

{ Consulting & Leadership }

Project Management (Agile)

Getting it delivered

Somebody accountable for scope, dates and the awkward conversations, so a project has one owner instead of five opinions.

Automated team collaboration

Automated team collaboration, password management, task and time management, and built a semi-automatic project-showcase system, raising team productivity.

Designed a time-management and reporting system

Designed a time-management and reporting system that remained in production use for years without significant modification.

Gathered and analyzed business requirements

Gathered and analyzed business requirements to translate into actionable features and user stories aligned with data governance standards.

This service is part of work listed elsewhere on this page:

Held 4 repositories to one history standard · Put the documentation under its own linter · Ran development as 51 written initiatives · Delivered analysis and regular progress reports · Saved 150 hours a month across 30 projects · Established a Technical Support department · Overhauled internal processes, saving 8,000 hours · Guided company projects through development · Improved team communication and collaboration · Led company growth from 4 to 14 employees · Managed 30 successful GIS projects to delivery · Managed 5,000 hours of map development · Estimated 4,000 hours saved, team from 2 to 18

{ Consulting & Leadership }

Product Strategy & Requirements

Working out what to build

Turning what the business wants into something a developer can build, and saying out loud what is not worth building.

Rewrote error messages to stop blaming users

Rewrote the application's error messages against a written tone standard after a refused sign-in blamed the user for mistyping when the provider actually required an app-specific password, and covered it with a test that names the provider.

Built a 22-linter commit gate

Built a commit gate of 22 one-line linters plus five that earn a paragraph, with no warning tier and no inline suppressions permitted, covering HTML, CSS, JavaScript, Python, YAML, Markdown, shell, links, spelling, secrets and typography.

Built the hiring and seafarer career workspaces

Built hiring and seafarer career workspaces; the recorded implementation snapshot contains 151 stored functions across 48 tables for vacancies, applications, certificates, rank progression and sea time.

Built the social layer: posts, feed and groups

Built the platform's social layer — posts, feed, groups, mentions, a follower graph and an occasions digest — on the same function-first data layer as the rest of the product.

Built a loyalty wallet, leagues and badges

Built the loyalty and reputation system — 67 functions over a 31-table ledger, with leagues, badges and a redemption shop — taking a row lock on the balance to close the double-spend window.

Built company ownership claims and adjudication

Built company ownership claims end to end — a user claims a company, an administrator adjudicates, and an approval rewrites the authorization graph that decides who is allowed to edit what.

Built the operator back-office for the platform

Built the operator back-office — around 40 admin routes and 128 components covering claims, moderation, feature flags, cache and diagnostics — so the platform can be run without database access.

Built Stripe and in-app purchase entitlements

Built the payments and entitlements layer — Stripe alongside Apple and Google in-app purchase — gating the directory, search and export through an 11-table access model checked on the server.

Designed the product's UI and UX end to end

Designed the product's UI and UX end to end — directory grids, dual card/table views, live requirement validators and breadcrumb navigation.

Developed a Data Analytics reporting system

Developed a Data Analytics reporting system, increasing quarterly software revenue by 400% through Python-based PDF reports.

Presented 200 UI/UX improvements

Presented 200 UI/UX improvements for the GIS map application, boosting revenue by 10x through enhanced software features.

Optimized forecasting and investment strategies

Optimized forecasting and investment strategies for 11 electricity grid operators, driving operational efficiency through data-driven GIS solutions.

Brand communication informed by audience response

Used market research and audience feedback to shape Engineer ApS brand communication and outreach.

A consistent identity across channels

Developed Engineer ApS brand identity across offline and online material, connecting the presentation with its engineering portfolio. The founder reports a 200x increase in brand recognition from this work.

This service is part of work listed elsewhere on this page:

Ran development as 51 written initiatives · Designed a time-management and reporting system · Gathered and analyzed business requirements · Managed 5,000 hours of map development · Led a GIS map application to 10x revenue growth

{ Consulting & Leadership }

Technical Documentation

Writing it down so the next person can follow

Handovers, runbooks and manuals that let your own staff operate the system without ringing whoever built it.

Made six collectors state what they skipped

Made six reporting collectors state the part of their input they never read — files that stayed shut, probes not installed, journal lines unrecognised — and fitted thresholds to 116 measured samples, after establishing that a blind collection and a healthy host render the same page.

Wrote a parser verifying every FFI call

Wrote a parser that reads the real 7,308-line C header and verifies every call site, every enum constant and that every pointer-owning class is final, after a hand-written placeholder header let calls to three removed functions compile, link and crash.

Built a cross-language figure check

Built a cross-language content check that fails when a translation drops a figure the English states, and when a language uses notation it does not use — finding two Danish descriptions missing a metric and sixteen Ukrainian spans quoting in the English style.

Passed 106 of 106 PDF/UA-1 profile checks

Recorded PDF/UA-1 validation across nine documents with 106 of 106 configured accessibility-profile checks passing, while the archival variant failed one of 146 rules.

Wrote a scope rule after a credential leak

Wrote a scope rule into the repository after a restructure carried another company's inventory, firewall allowances and prose into it — and kept the quarantined residue under the secret scanner rather than excluding it.

Built 7 read-only host reporting roles

Built seven read-only reporting roles that render a live host to Markdown — facts, access, git, metrics, traffic, security and provider inventory — under a rule that no number is printed the run did not measure.

Ran a continuous security scanning programme

Established a continuous security programme — code scanning, DAST, dependency and vulnerability checks, SBOM generation, secret scanning and SHA-pinned actions — alongside 21 written security audits.

Authored 578 go-task automation targets

Authored 578 go-task automation targets spanning native, Docker and HTTPS dev modes, linting, testing, database and deployment.

Migrated the API to Huma v2 at 100% parity

Migrated the HTTP API from Fiber to Huma v2 — 649 paths and 760 operations — reaching and holding 100% parity between the routes the server registers and the OpenAPI description it publishes.

Delivered 8 Power BI projects with manuals

Delivered 8 Power BI projects with comprehensive manuals, integrating Microsoft Power BI tools with NodeJS API and Python FastAPI for effective data analytics and visualization.

This service is part of work listed elsewhere on this page:

Held 4 repositories to one history standard · Rewrote error messages to stop blaming users · Found the hooks and the gate disagreeing · Put the documentation under its own linter · Ran development as 51 written initiatives · Built a 22-linter commit gate · Built a 268-check guard engine for the repo · Wrote 50,000 words of software documentation · One infrastructure plan for 14 departments

Compare your starting point

Choose a comparable piece of work and identify what differs in your own situation.

See what it costs Get in touch →

Discuss a relevant approach

Ask us to turn the relevant experience into an approach you can evaluate.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

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

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://advisory.engineer.company/portfolio/