Engineer

{ 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.

Er I stadig ved at finde ud af, hvad I har brug for?

Fortæl os, hvad der går galt, eller hvad I prøver at bygge. Så siger vi, hvad det vil kræve, cirka hvad det vil koste, og om vi er de rigtige til det.

Fra 1.400 DKK i timen, ekskl. moms — og mindre pr. time ved en længere booking.

Kontakt os Se, hvad det koster

Udvalgte resultater, der demonstrerer denne ydelse.

Gør ydelsen til et resultat

Lad »Frontend-udvikling« understøtte et forretningsmål. Vi forbinder det tekniske arbejde med de brugere, den drift og de beslutninger, det skal støtte. Brugerflader, der indlæses hurtigt, virker på en telefon og opfører sig ens i alle de browsere, jeres kunder nu bruger.

Kontakt os

Undersøg det relevante arbejde

Brug arbejdet på siden til at undersøge vores tilgang. Spørg til afvejningerne og forudsætningerne bag hvert resultat. Har styret full-stack-udviklingen af GIS-kort — full stack-kortløsning, fra database til browser

Har styret full-stack-udviklingen af GIS-kort Kontakt os →

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 præsenteret 200 UI/UX-forbedringer

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

Har bidraget til design af brugergrænseflader

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

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 designet 30 websites fra Photoshop til HTML

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

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 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 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 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 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 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 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 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 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 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 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.

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 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.

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.

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.

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.

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 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 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 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 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 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.

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 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 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.

Leder I efter Frontend-udvikling?

Brug de tilknyttede opgaver til at vurdere relevant erfaring. Fortæl os, hvad I skal bygge; vi kan tale om omfang, fremgangsmåde og overdragelse.

Kontakt os Hele porteføljen

Afgræns opgaven før bestilling

Beskriv arbejdsgangen, de eksisterende grænseflader og de brugere, der har brug for ændringen. Vi kan definere en testbar leverance og en overdragelse, der gør teamet i stand til at vedligeholde løsningen.

Se, hvad det koster Kontakt os →

Gennemgå jeres krav med os

Fortæl os, hvad I har brug for inden for »Frontend-udvikling«, herunder fristen og det team, der skal vedligeholde løsningen. Vi kan drøfte en praktisk vej frem.

Kontakt os

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/services/frontend-development/