Engineer Programvare

Pålitelighet og sikkerhetskopier

21 resultater

Knytt sammen delene av systemet

Arbeidet under «Pålitelighet og sikkerhetskopier» viser én del av et fungerende system. Vi kan hjelpe med å knytte den delen til resten av driften deres. Alt som skjer på et sted — fartøy, kjøretøy, tomter, sensorer — lagt på et kart du kan søke i, måle på og dele.

Se tjenestene våre Ta kontakt →

Finn en sammenlignbar utfordring

Åpne et eksempel som ligner situasjonen deres. Gjennomgangen forklarer beslutningene bak resultatet, utover navnet på teknologien. Gjorde en geodatapipeline 50 ganger raskere — 50x raskere pipelines for geografiske data

Gjorde en geodatapipeline 50 ganger raskere Ta kontakt →

Drev 100 datavarehus: Postgres, MS SQL, BigQuery

Utformet, opprettet og administrerte 100 databaser for datavarehus i PostgreSQL, MS SQL og Google BigQuery, hovedsakelig med GIS- og tidsseriedata, og optimaliserte ytelse og skalerbarhet.

Automatiserte GIS-SaaS-utrulling i GitHub Actions

Automatiserte produksjonssettingen, databehandlingen og rapporteringssystemet for en GIS SaaS-applikasjon med GitHub Actions CI/CD, Python, Bash og SQL.

Automatiserte leveransen av 20 GIS-datapipelines

Automatiserte leveransen av 20 GIS-datapipelines og ETL-prosesser for applikasjonsdata, og effektiviserte infrastrukturautomatisering og rapportering.

Ledet en GIS-kartapp til 10 ganger mer omsetning

Ledet programvareutviklingen av en GIS-kartapplikasjon, økte inntektene 10 ganger og posisjonerte produktet som en primær dataressurs.

Ledet 30+ GIS-prosjekter fra utvikling til støtte

Ledet utvikling, produksjonssetting og støtte for over 30 GIS-prosjekter, med ekspertise i teknologiene PostgreSQL, Bash, Python, JavaScript, GDAL, ArcGIS, PostGIS og Mapbox.

Administrerte 40 nettsteder på Apache og Nginx

Administrerte 40 nettsteder på Ubuntu Linux-servere for webhotell med Apache og Nginx, og sikret høy tilgjengelighet og ytelse.

Et kartprodukt, håndtert fra ende til annen

Utformet, utviklet og støttet infrastruktur, databehandling og en kartapplikasjon over 2 år, med både programvareleveranser og driftsvedlikehold.

Genererte API-kontrakten med databasen som kilde

Genererte API-kontrakten fra databasen og utover — OpenAPI, en typet TypeScript-klient på 44 076 linjer, 61 mock-handlere og grensene brukergrensesnittet håndhever — med en kontroll i hvert ledd som feiler ved avvik.

Bygde e-post med failover og leveringslogging

Bygde e-post som en plattformfunksjon — tre leverandører med failover, leveringswebhooks, logging av sending og levering, maler og kampanjer — bak en oppstartskontroll som ikke lar plattformen starte uten minst én leverandør.

Leverte full støtte for Progressive Web App

Leverte full støtte for Progressive Web App — kan installeres og fungerer uten nettforbindelse — med caching under kjøring i Workbox via next-pwa.

Bygde infrastruktur som fortsatt kjører år senere

Planla og implementerte ny infrastrukturfunksjonalitet for interne og eksterne systemer, og bygde løsninger så solide at de fortsatt kjører år senere med minimale endringer.

Bygde en testsuite for Go, SQL og nettleser

Bygde en lagdelt automatisert testsuite — 981 Go-tester, 543 spesifikasjoner for frontend og nettleser, 494 atferdstester i SQL — med mutasjonstesting, egenskapsbaserte tester og en kvalitetsport for universell utforming.

Flyttet tregt arbeid over i en River-jobbkø

Flyttet tregt arbeid ut av forespørselsbanen og over i en River-jobbkø — 15 worker-moduler, 8 planlagte oppgaver og 20 pg_cron-jobber — slik at en forespørsel returnerer mens arbeidet bak den fortsetter.

Bygde egen feilovervåking og sporing

Bygde egen feilovervåking og OpenTelemetry-sporing i stedet for å kjøpe dem — rensing av nyttelast, oppdagelse av topper og regresjoner, symbolisering og et syntetisk hjerteslag — bak 11 operatørvisninger.

Bygde fail-closed hastighetsgrenser mot misbruk

Bygde misbruksvern som stenger ved feil — 22 Redis-baserte hastighetsbegrensere, Cloudflare Turnstile, idempotente forespørsler og en origin-lås — slik at plattformen avviser boter og flom i stedet for å stole på dem som kaller den.

Offentlige tjenester på en liten server

Driftet selskapets offentlige tjenester på en vert med én kjerne og 512 MB — en git-forge, en webserver for sju domener, Tor, to servere for alternative protokoller og utestenging av inntrengere — med 464 MB brukbart minne som designbegrensning; backupautomatisering er en separat implementasjon.

Holdt generatoren til 981 tester og 92 % dekning

Holdt generatoren til 981 testtilfeller med et minstekrav på 92 % grendekning og advarsler behandlet som feil, og verifiserte idempotens ved å kjøre hele byggepipelinen to ganger fra en tom fil og kreve at den andre kjøringen ikke endret noe.

En avgrenset hendelsesflyt i en macOS-app

Avgrenset hendelsesbuffere og logging etter en registrert macOS-hendelse med vekst på 41 MB i sekundet og 111 GB komprimerte sider på en maskin med 36 GB; en registrert full testkjøring reduserte kjernens logglinjer fra 610 996 til 1 411.

Tok i bruk streng samtidighet i Swift 6 over C

Tok i bruk fullstendig streng samtidighet i Swift 6 uten actors, og koblet en blokkerende C-hendelsesløkke til main actor gjennom én produsent, én konsument og én rekkefølge — etter å ha slått fast at én task per hendelse mister rekkefølgen grensesnittet er avhengig av.

Nådde den ubrukte halvdelen av kjernen

Tok i bruk den halvdelen av meldingskjernen applikasjonen aldri hadde brukt — overføring av sikkerhetskopier, forsvinnende meldinger, redigering og ny sending av meldinger, verifiserte invitasjoner, proxyer og krypteringspolicy — og kjørte hver test mot det ekte biblioteket uten mocks.

Fikk seks innsamlere til å si hva de hoppet over

Fikk seks rapporteringsinnsamlere til å oppgi den delen av inndataene de aldri leste — filer som forble lukket, sonder som ikke var installert, journallinjer som ikke ble gjenkjent — og tilpasset terskelverdier til 116 målinger, etter å ha slått fast at en blind innsamling og en vert der alt er i orden, gir den samme siden.

Fokuser den tekniske innsatsen

Beskriv arbeidsflyten, eksisterende grensesnitt og brukerne som trenger endringen. Vi kan definere en testbar leveranse og en overlevering som gjør teamet i stand til å vedlikeholde løsningen.

Se hva det koster Ta kontakt →

Sett emnet i deres sammenheng

Fortell hvordan «Pålitelighet og sikkerhetskopier» inngår i systemet deres. Vi kan hjelpe med å velge riktig tjeneste og avgrense arbeidet.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://software.engineer.company/nb/categories/reliability/