Engineer

Dokumentasjon

38 resultater

Knytt sammen delene av systemet

Arbeidet under «Dokumentasjon» viser én del av et fungerende system. Vi kan hjelpe med å knytte den delen til resten av driften deres. Overleveringsdokumenter, driftsinstrukser og håndbøker som lar dine egne ansatte drifte systemet uten å ringe den som bygde det.

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. Én infrastrukturplan for 14 avdelinger — 14 forskningsavdelinger på én felles infrastrukturplan

Én infrastrukturplan for 14 avdelinger Ta kontakt →

Én infrastrukturplan for 14 avdelinger

Designet et helhetlig infrastrukturrammeverk for et forskningsinstitutt som berørte 14 avdelinger, med fleksible moduler, samlede datapipelines og strukturerte støttestrategier for varig innføring.

Leverte 8 Power BI-prosjekter med veiledninger

Leverte 8 Power BI-prosjekter med grundige brukerveiledninger, og integrerte Microsoft Power BI-verktøy med et NodeJS-API og Python FastAPI for effektiv dataanalyse og visualisering.

Publiserte 500 datarapporter om strøm og GIS

Publiserte 500 analyserapporter om strøm- og GIS-data, basert på grundige undersøkelser og feilsøking, for å sikre presis innsikt i geografiske stordata og tidsseriedata.

Skrev 1 000 RBAC-regler for å sikre Linux-servere

Styrket datasikkerheten ved å innføre 1 000 RBAC-regler for utviklere, applikasjonsinstanser, PostgreSQL, MS SQL og andre Linux-servere, og forhindret uautorisert tilgang; dokumentert med Ansible-automatisering.

Skrev 50 000 ord programvaredokumentasjon

Skrev 50 000 ord grundig programvaredokumentasjon i Markdown i GitHub, Craft og Confluence, og sikret dermed at kunnskapen ble bevart og prosessen gjennomsiktig.

Samlet inn og analyserte forretningskrav

Samlet inn og analyserte forretningskrav og omsatte dem til gjennomførbare funksjoner og brukerhistorier i tråd med standardene for datastyring.

Forbedret kommunikasjonen og samarbeidet i teamet

Forbedret kommunikasjonen og samarbeidet i teamet ved å innføre Slack, Mattermost, 1Password og Jira, og sparte 8 000 arbeidstimer.

Migrerte API-et til Huma v2 med 100 % paritet

Migrerte HTTP-API-et fra Fiber til Huma v2 — 649 stier og 760 operasjoner — og nådde og holdt 100 % samsvar mellom rutene serveren registrerer og OpenAPI-beskrivelsen den publiserer.

Skrev 578 automatiseringsmål i go-task

Skrev 578 go-task-automatiseringsmål som dekker native-, Docker- og HTTPS-utviklingsmoduser, linting, testing, database og produksjonssetting.

Designet «Liquid Glass»-designsystem for iOS 26

Designet et «Liquid Glass»-designsystem for iOS 26 og et kanonisk komponentregister håndhevet av lint-regler for å hindre at brukergrensesnittet spriker.

Bygde et engelsk/dansk i18n-system

Etablerte et engelsk/dansk internasjonaliseringssystem med 12 027 meldinger per språkversjon fordelt på 458 navneromsfiler, med vokabular håndhevet av linter og et budsjett på 400 linjer per fil.

Satte et krav om null advarsler på seks språk

Satte en kvalitetsstandard med null advarsler for seks språk — Go, TypeScript, SQL, Python, Shell og Markdown — håndhevet av pre-commit-hooks.

Bygde repoets kontrollmotor med 268 kontroller

Bygde repoets kontrollmotor — 268 registrerte commit-kontroller, 277 lint-regler og 15 egne ESLint-regler — pluss 146 tester av selve kontrollene, slik at det er bygget, ikke gjennomgangen, som holder standarden.

Drev et løpende program for sikkerhetsskanning

Etablerte et kontinuerlig sikkerhetsprogram — kodeskanning, DAST, kontroll av avhengigheter og sårbarheter, SBOM-generering, skanning etter hemmeligheter og SHA-låste actions — sammen med 21 skriftlige sikkerhetsrevisjoner.

Infrastruktur beskrevet i et kodelager

Bygde selskapets egen infrastruktur som 19 Ansible-playbooks og 34 roller på 12 065 linjer YAML, som konvergerer en vert i drift mot en deklarert tilstand der hver play er idempotent.

Backupautomatisering med en tydelig gjenopprettingsmetode

Implementerte restic-backupautomatisering med lagringsregler, integritetskontroll og en arbeidsflyt for månedlig gjenopprettingstest; dokumenterte aktivering av lagring og kontroll av en virkelig gjenoppretting som separate driftsoppgaver.

Bygde 7 skrivebeskyttede rapportroller for verter

Bygde sju skrivebeskyttede rapporteringsroller som gjengir en vert i drift som Markdown — fakta, tilgang, git, metrikker, trafikk, sikkerhet og leverandørens inventar — etter en regel om at ingen tall skrives ut som kjøringen ikke har målt.

Skrev en avgrensningsregel etter tilgangslekkasje

Skrev en avgrensningsregel inn i repoet etter at en omstrukturering hadde ført et annet selskaps inventar, brannmuråpninger og tekst inn i det — og holdt de karantenesatte restene under skanneren for hemmeligheter i stedet for å unnta dem.

Bygde en commit-port med 22 lintere

Bygde en commit-port med 22 enlinjes lintere pluss fem som fortjener et avsnitt, uten advarselsnivå og uten tillatte undertrykkinger i koden, som dekker HTML, CSS, JavaScript, Python, YAML, Markdown, shell, lenker, staving, hemmeligheter og typografi.

Byttet 27 skriftstørrelser mot en skala i 6 trinn

Erstattet 27 gjengitte skriftstørrelser, der de nærmeste naboene lå 0,6 % fra hverandre, med en Major Third-skala i seks trinn, og skrev linteren som avviser den tjueåttende.

Tilgjengelighetskontroller på 32 sider i begge temaer

Implementerte WCAG 2.2 Level AA-kontroller på 32 representative sider i begge fargetemaer, med en samsvarsoversikt, kontroll av ett AAA-kriterium og det delvise resultatet for kriterium 1.4.13 uttrykkelig angitt.

Lukket fargesystemet ved 28 farger

Lukket fargesystemet ved 28 dokumenterte farger med en linter som avviser en verdi som er malt, men udokumentert, dokumentert, men ikke malt, feilmålt, skrevet om som en literal, eller innenfor en perseptuell avstand på 0,02 fra en som allerede finnes.

Bygde et utskriftsstilark på 637 linjer

Bygde et utskriftsstilark på 637 linjer mot fire dokumenterte atferder i gjengivelsesmotorer, blant dem innfelte overskrifter som ble skrevet ut hvitt på hvitt for enhver leser med nettleseren innstilt på mørkt.

Drev utviklingen som 51 skriftlige initiativer

Drev utviklingen av nettstedet som 51 skriftlige initiativer fordelt på 54 plandokumenter, hvert med sine åpne avkrysningsbokser, sine målinger og beslutningene det avslo — blant dem seks av åtte funn om agentberedskap som ble avvist med begrunnelsen registrert.

La dokumentasjonen under sin egen linter

Satte dokumentasjonen under sin egen linter — et linjebudsjett per fil som bare kan strammes inn, et krav om indeks og en kontroll av reelle stier — som fant seks av ni planer merket som ferdige med uavkryssede bokser, og fjorten commit-sperrende oppgaver som ikke var nevnt i noe dokument.

Lintet selve kvalitetsporten på 10 242 linjer

Satte 10 242 linjer JavaScript i kvalitetsporten under en formaterer og en linter etter å ha slått fast at det var den største kodemengden i repoet og den eneste ingenting leste, og rettet 13 funn uten å undertrykke noen.

Bestod 106 av 106 PDF/UA-1-profilkontroller

Dokumenterte PDF/UA-1-validering av ni dokumenter med 106 av 106 bestått kontroller i den konfigurerte profilen for universell utforming, mens arkivvarianten feilet på én av 146 regler.

Aktiverte alle linterregler som feil

Slo på alle reglene i Python-linteren på feilnivå og arbeidet seg gjennom 1 815 funn ned til null, med en skriftlig begrunnelse for hvert av de få unntakene og to av dem sikret av en kontroll i stedet for en kommentar.

Bygde inn en QR-enkoder på 522 linjer

Bygde inn en QR-enkoder — Reed-Solomon over GF(256), fast moduloppsett, åtte maskemønstre — på 522 linjer i stedet for å ta inn en tredje kjøretidsavhengighet, og verifiserte den ved å lese den ferdige matrisen tilbake med en uavhengig skrevet dekoder.

Fant at hookene og porten var uenige

Fant ut at commit-hookene og kvalitetsporten kjørte ulike kontroller mens et dokument lovet at de var de samme, ved å sammenligne de to listene i en test — de fem som bare noen gang ble kjørt manuelt, var de som leste CV-teksten.

Flyttet hver tekststreng inn i et innholdstre

Flyttet alle brukerrettede strenger ut av Python og inn i et innholdstre på 874 filer på tre språk, etter å ha funnet døde oversettelser ingen kunne se var døde, og en kontroll som i det stille vurderte en tredjedel av resultatene.

Bygde en tallkontroll på tvers av språk

Bygde en innholdskontroll på tvers av språk som feiler når en oversettelse mister et tall den engelske teksten oppgir, og når et språk bruker en notasjon som ikke hører hjemme i det — og fant to danske beskrivelser som manglet en måleverdi, og seksten ukrainske tekstbiter med engelske anførselstegn.

Skrev en parser som verifiserer hvert FFI-kall

Skrev en parser som leser den faktiske C-headeren på 7 308 linjer og verifiserer hvert kallsted, hver enum-konstant og at hver klasse som eier en peker, er final, etter at en håndskrevet plassholder-header lot kall til tre fjernede funksjoner kompilere, lenke og krasje.

Ga alle 17 ikoner navn for skjermlesere

Ga hver kontroll med bare ikon i grensesnittet et navn for skjermlesere etter å ha funnet at send-knappen ble lest opp som «arrow up circle, button», og skrev linteren som krever etiketten innen åtte linjer fra ikonet.

Skrev om feilmeldingene som skyldte på brukerne

Skrev om applikasjonens feilmeldinger i tråd med en skriftlig tonestandard etter at en avvist innlogging ga brukeren skylden for en skrivefeil når leverandøren egentlig krevde et appspesifikt passord, og dekket det med en test som navngir leverandøren.

Kjenn lisensen til alt dere leverer

Reviderte 644 Rust-crates for lisenskompatibilitet ved hvert bygg, og beviste at kontrollen slår ut ved å skrive om lisensen til én crate og ved å flytte den låste kjernerevisjonen uten å regenerere.

Holdt 4 repoer til én historikkstandard

Holdt fire repoer til én historikkstandard — konvensjonell, uten emojier og uten attribusjonslinjer, håndhevet av en hook for commit-meldinger — sammen med 48 instruksjonsdokumenter som styrer hvordan arbeidet gjøres.

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

Ta med beslutningen, menneskene som berøres og alternativene som er vurdert. Vi kan tydeliggjøre avveiningene og etterlate en dokumentert anbefaling.

Se hva det koster Ta kontakt →

Sett emnet i deres sammenheng

Fortell hvordan «Dokumentasjon» 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://engineer.company/nb/categories/documentation/