Engineer Rådgivning

Test & QA

21 resultater

Forbind delene af jeres system

Arbejdet under »Test & QA« viser en del af et fungerende system. Vi kan hjælpe med at forbinde den del med resten af jeres drift. Testet, udgivet og til at rulle tilbage, så ændringer når kunderne en tirsdag eftermiddag i stedet for sent en fredag aften.

Se vores ydelser Kontakt os →

Find en sammenlignelig udfordring

Åbn et eksempel, der ligner jeres situation. Gennemgangen beskriver beslutningerne bag resultatet, ud over teknologiens navn. Har fornyet processer og sparet 8.000 timer — 8,000 timer sparet ved at lægge arbejdsgangene om

Har fornyet processer og sparet 8.000 timer Kontakt os →

Har strømlinet CI/CD og sparet 4.000 timer

Har strømlinet CI/CD-processer og sparet 4.000 timer ved at indføre automatisering i softwareudviklingspipelines.

Har migreret API'et til Huma v2 med 100 % paritet

Har migreret HTTP-API'et fra Fiber til Huma v2 — 649 paths og 760 operationer — og opnået og fastholdt 100 % paritet mellem de ruter, serveren registrerer, og den OpenAPI-beskrivelse, den udgiver.

Har skrevet 578 go-task-automatiseringsmål

Har skrevet 578 go-task-automatiseringsmål, der dækker udviklingstilstandene native, Docker og HTTPS, linting, test, database og udrulning.

Har indført nul advarsler som krav i seks sprog

Har sat en kvalitetsstandard med nul advarsler i seks sprog — Go, TypeScript, SQL, Python, Shell og Markdown — håndhævet af pre-commit-hooks.

Har bygget en testsuite til Go, SQL og browseren

Har bygget en lagdelt automatiseret testsuite — 981 Go-tests, 543 frontend- og browserspecs, 494 SQL-adfærdstests — med mutationstest, property-based tests og en tilgængelighedsgate.

Har bygget en kontrolmotor med 268 tjek

Har bygget repositoryets kontrolmotor — 268 registrerede commit-tjek, 277 lint-regler og 15 egne ESLint-regler — plus 146 tests af selve tjekkene, så det er builden, der holder standarden, ikke reviewet.

Har bygget en commit-gate med 22 lintere

Har bygget en commit-gate med 22 lintere, der hver kan beskrives på én linje, plus fem, der fortjener et afsnit, uden advarselsniveau og uden tilladte inline-undertrykkelser, og som dækker HTML, CSS, JavaScript, Python, YAML, Markdown, shell, links, stavning, hemmeligheder og typografi.

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 sat dokumentationen under sin egen linter

Har sat dokumentationen under sin egen linter — et linjebudget pr. fil, der kun kan sænkes, et indekskrav og et tjek af faktiske stier — som fandt seks af ni planer markeret som færdige, selvom de stadig havde felter, der ikke var krydset af, og fjorten opgaver, der spærrede hver commit uden at være nævnt i noget dokument.

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 givet generatoren 981 tests og 92 % dækning

Har holdt generatoren på 981 testtilfælde og en nedre grænse på 92 % grendækning med advarsler behandlet som fejl og eftervist idempotens ved at køre hele build-pipelinen to gange fra en tom fil og kræve, at det andet gennemløb intet ændrer.

Har bestået 106 af 106 PDF/UA-1-profiltjek

Har dokumenteret PDF/UA-1-validering af ni dokumenter med 106 af 106 beståede kontroller i den konfigurerede tilgængelighedsprofil, mens arkivvarianten fejlede på én af 146 regler.

Har slået alle linterregler til som fejl

Har slået hver eneste regel i Python-linteren til som fejl og arbejdet 1.815 fund ned til nul, hvor hver af de få undtagelser har en skriftlig begrundelse, og to af dem er sikret af et tjek i stedet for en kommentar.

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 fundet uenighed mellem hooks og kvalitetsgate

Har opdaget, at commit-hooks og kvalitetsgaten kørte forskellige tjek, selvom et dokument lovede, at de var de samme, ved at sammenligne de to lister i en test — de fem, der kun nogensinde blev kørt manuelt, var dem, der læste CV-prosaen.

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 indført Swift 6 strict concurrency over C

Har indført Swift 6's complete strict concurrency uden actors og bygget bro fra en blokerende C-hændelsesløkke til main actor gennem én producent, én forbruger og én rækkefølge — efter at have fastslået, at en opgave pr. hændelse mister den rækkefølge, grænsefladen afhænger af.

Har skrevet en parser, der tjekker hvert FFI-kald

Har skrevet en parser, der læser den rigtige C-header på 7.308 linjer og verificerer hvert kaldssted, hver enum-konstant, og at hver klasse, der ejer en pointer, er final, efter at en håndskrevet pladsholder-header lod kald til tre fjernede funktioner kompilere, linke og crashe.

Har fået testsuiten ned på 8,9 sekunder

Har taget testsuiten fra seks tests over en grænse på tres sekunder til 135 beståede på 8,9 sekunder ved at profilere hovedtråden og fjerne de to kald, den hang i ved 3.989 ud af 4.017 stikprøver.

Har renset fejlbeskederne for bebrejdelser

Har omskrevet applikationens fejlbeskeder efter en nedskrevet standard for tone, efter at et afvist login gav brugeren skylden for at have tastet forkert, mens udbyderen i virkeligheden krævede en applikationsspecifik adgangskode, og dækket det med en test, der nævner udbyderen.

Har fået seks indsamlere til at melde det ulæste

Har fået seks rapporteringsindsamlere til at oplyse den del af deres input, de aldrig læste — uåbnede filer, ikke-installerede sonder og ugenkendte journallinjer — og tilpasset tærskler til 116 målte stikprøver efter at have fastslået, at en blind indsamling og en rask vært giver den samme side.

Sæt fokus på den tekniske indsats

Medbring udrulningsforløbet, driftsrisiciene og dem, der har vagten. Vi kan afgrænse en ændring med kontroller, en vej tilbage og tydeligt ansvar.

Se, hvad det koster Kontakt os →

Sæt emnet ind i jeres sammenhæng

Fortæl os, hvordan »Test & QA« indgår i jeres system. Vi kan hjælpe med at vælge den relevante ydelse og afgrænse arbejdet.

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://advisory.engineer.company/da/categories/testing/