Engineer Консультування

Тестування та QA

21 досягнення

Поєднайте частини системи

Роботи за темою «Тестування та QA» показують одну частину дієвої системи. Ми можемо допомогти пов'язати її з рештою вашої діяльності. Перевірено, випущено і з можливістю відкату, тож зміни доходять до клієнтів у вівторок удень, а не пізно в п'ятницю.

Переглянути наші послуги Зв'язатися з нами →

Знайдіть схоже завдання

Відкрийте приклад, схожий на вашу ситуацію. Огляд пояснює рішення, що привели до результату, а не лише називає технологію. Реорганізувала процеси, заощадивши 8 000 годин — 8,000 годин заощаджено завдяки перебудові робочих процесів

Реорганізувала процеси, заощадивши 8 000 годин Зв'язатися з нами →

Оптимізувала CI/CD, заощадивши 4 000 годин

Оптимізувала процеси CI/CD, заощадивши 4 000 годин завдяки впровадженню автоматизації в пайплайнах розробки програмного забезпечення.

Перенесла API на Huma v2 зі 100% відповідністю

Перенесла HTTP API з Fiber на Huma v2 — 649 шляхів і 760 операцій — досягнувши та втримавши 100% відповідність між маршрутами, які реєструє сервер, і описом OpenAPI, який він публікує.

Написала 578 цілей автоматизації go-task

Написала 578 цілей автоматизації go-task, що охоплюють нативний, Docker- та HTTPS-режими розробки, лінтинг, тестування, базу даних та розгортання.

Запровадила нуль попереджень у шести мовах

Встановила стандарт якості без жодних попереджень для шести мов — Go, TypeScript, SQL, Python, Shell і Markdown — дотримання якого забезпечували pre-commit hooks.

Побудувала тести для Go, SQL і браузера

Побудувала багаторівневий набір автоматизованих тестів — 981 тест Go, 543 фронтендні та браузерні специфікації, 494 поведінкові тести SQL — з мутаційним тестуванням, property-based тестами та обов'язковою перевіркою доступності.

Побудувала механізм із 268 перевірок репозиторію

Побудувала механізм перевірок репозиторію — 268 зареєстрованих перевірок коміту, 277 правил лінтингу та 15 власних правил ESLint — а також 146 тестів самих перевірок, щоб стандарт тримала збірка, а не код-рев'ю.

Побудувала перевірку комітів із 22 лінтерів

Побудувала перевірку перед комітом із 22 лінтерів, кожен з яких описується одним рядком, і ще п'ятьох, що заслуговують окремого абзацу, без рівня попереджень і без дозволу на вбудоване вимкнення правил, яка охоплює HTML, CSS, JavaScript, Python, YAML, Markdown, shell, посилання, орфографію, секрети й типографіку.

Страховка, що більше не забирає пів години

Скоротила браузерну перевірку якості сайту з 1 636 секунд до 615, запускаючи перевірки від найдовших до найкоротших через пул воркерів із максимум чотирма одночасними перевірками, після того як виміряла, що алфавітний порядок коштував 320 секунд проти 224.

Підключила до документації окремий лінтер

Підключила до документації окремий лінтер — ліміт рядків для кожного файлу, який можна лише знижувати, вимогу індексу та перевірку справжніх шляхів, — який знайшов шість із дев'яти планів, позначених виконаними, попри невідмічені пункти, і чотирнадцять завдань перевірки перед комітом, не згаданих у жодному документі.

Перевірила лінтером 10 242 рядки перевірки якості

Охопила форматувальником і лінтером 10 242 рядки JavaScript у перевірках якості, з'ясувавши, що це найбільший обсяг коду в репозиторії і єдиний, який ніщо не перевіряло, — виправила 13 зауважень і не вимкнула жодного.

Забезпечила генератору 981 тест і 92% покриття

Покрила генератор 981 тестовим випадком із мінімальним порогом покриття гілок 92%, де попередження вважаються помилками, і перевірила ідемпотентність, двічі запустивши весь пайплайн збірки з порожнього файлу та вимагаючи, щоб другий прохід нічого не змінював.

PDF/UA-1: пройдено 106 зі 106 перевірок профілю

Зафіксувала перевірку PDF/UA-1 для 9 документів: успішно пройдено 106 зі 106 налаштованих перевірок профілю доступності, тоді як архівний варіант не пройшов одне зі 146 правил.

Увімкнула всі правила лінтера Python як помилки

Увімкнула всі правила лінтера Python як помилки й довела 1 815 зауважень до нуля; кожен із небагатьох винятків має письмове обґрунтування, а два з них підкріплені перевіркою, а не коментарем.

Написала тести для самих перевірок

Написала тести для самих перевірок, з'ясувавши, що перевірка, яка отримує лише чисті вхідні дані, одного дня звітуватиме про чистоту, бо нічого не прочитала: підклала помилку правопису, щоб підтвердити, що перевірка орфографії її знаходить, і взяла діапазон ідентифікаторів із бази даних, а не з числа в тесті.

Виявила розбіжність між хуками й гейтом якості

Виявила, що хуки комітів і гейт якості виконують різні перевірки, хоча документ обіцяв, що вони однакові, порівнявши два переліки в тесті, — п'ять перевірок, які запускалися лише вручну, були саме тими, що читали текст CV.

Побудувала міжмовну перевірку чисел

Побудувала міжмовну перевірку контенту, яка падає, коли переклад губить число, вказане англійською, або коли мова вживає не притаманну їй нотацію, — і знайшла два данські описи без метрики та шістнадцять українських фрагментів, що цитували на англійський лад.

Увімкнула сувору конкурентність Swift 6 поверх C

Перейшла на повну сувору перевірку конкурентності Swift 6 без акторів, з'єднавши блокувальний цикл подій C з головним актором через одного виробника, одного споживача й один порядок, — з'ясувавши, що окрема задача на кожну подію губить порядок, від якого залежить інтерфейс.

Написала парсер, що перевіряє кожен виклик FFI

Написала парсер, який читає справжній C-заголовок на 7 308 рядків і перевіряє кожне місце виклику, кожну константу переліку й те, що кожен клас-власник вказівника є final, після того як через написаний вручну заголовок-заглушку виклики трьох вилучених функцій компілювалися, лінкувалися й аварійно завершувалися.

Пришвидшила набір тестів до 8,9 секунди

Скоротила набір тестів від шести тестів, що перевищували ліміт у шістдесят секунд, до 135 успішних за 8,9 секунди, профілюючи головний потік і прибравши два виклики, у яких він перебував 3 989 із 4 017 вибірок.

Переписала помилки, щоб не винити користувачів

Переписала повідомлення про помилки застосунку за письмовим стандартом тону після того, як у разі відмови у вході застосунок звинувачував користувача в помилці введення, хоча провайдер насправді вимагав пароль для застосунку, і покрила це тестом, який називає провайдера.

Змусила шість збирачів звітувати про пропущене

Змусила шість збирачів звітів повідомляти ту частину свого входу, яку вони так і не прочитали, — файли, що лишилися закритими, невстановлені зонди, нерозпізнані рядки журналу, — і підігнала пороги до 116 виміряних зразків, з'ясувавши, що сліпий збір і справний хост дають однакову сторінку.

Спрямуйте інженерні зусилля

Опишіть шлях розгортання, експлуатаційні ризики та чергову команду. Ми можемо визначити зміну з перевірками, шляхом відновлення та чіткою відповідальністю.

Дивитися, скільки це коштує Зв'язатися з нами →

Розгляньте тему у своєму контексті

Розкажіть, як «Тестування та QA» стосується вашої системи. Ми допоможемо обрати відповідну послугу та визначити обсяг роботи.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

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

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://advisory.engineer.company/uk/categories/testing/