Engineer Софтвер

{ DevOps та автоматизація CI/CD }

Випускати зміни без нервів

Перевірено, випущено і з можливістю відкату, тож зміни доходять до клієнтів у вівторок удень, а не пізно в п'ятницю.

Ще визначаєтесь, що саме вам потрібно?

Розкажіть, що йде не так або що ви намагаєтесь побудувати. Ми скажемо, що для цього потрібно, скільки це приблизно коштуватиме і чи ми ті люди, які вам потрібні.

Від 1 400 DKK за годину, без ПДВ — і дешевша година на довшому замовленні.

Зв'язатися з нами Дивитися, скільки це коштує

Вибрані роботи, що демонструють цю послугу.

Перетворіть послугу на результат

Нехай «DevOps та автоматизація CI/CD» слугує меті бізнесу. Ми можемо пов'язати технічну роботу з користувачами, експлуатацією та рішеннями, які вона має підтримувати. Перевірено, випущено і з можливістю відкату, тож зміни доходять до клієнтів у вівторок удень, а не пізно в п'ятницю.

Зв'язатися з нами

Розгляньте відповідні роботи

Використайте роботи на цій сторінці, щоб оцінити наш підхід. Запитайте про компроміси та передумови кожного результату. Автоматизувала розгортання GIS SaaS через CI/CD — розгортання, дані та звітність — усе без ручної роботи

Автоматизувала розгортання GIS SaaS через CI/CD Зв'язатися з нами →

Автоматизувала розгортання GIS SaaS через CI/CD

Автоматизувала розгортання GIS SaaS-застосунку, обробку даних та систему звітності за допомогою GitHub Actions CI/CD, Python, Bash та SQL.

Автоматизувала розгортання 20 GIS-пайплайнів

Автоматизувала розгортання 20 пайплайнів GIS-даних та ETL-процесів даних застосунку, оптимізувавши автоматизацію інфраструктури та звітність.

Реорганізувала процеси, заощадивши 8 000 годин

Реорганізувала внутрішні процеси, заощадивши 8 000 годин завдяки вдосконаленню архітектури програмного забезпечення, систем та ефективності планування.

Спрощена інфраструктура поза AWS

Перебудувала інфраструктуру міжнародного клієнта, вилучивши невикористані служби, скоригувавши потужність і перенісши навантаження з AWS. За свідченням засновниці, витрати на хмару скоротилися у 10 разів без втрати продуктивності.

Згенерувала контракт API із самої бази даних

Згенерувала контракт API з бази даних назовні — OpenAPI, типізований TypeScript-клієнт на 44 076 рядків, 61 mock-обробник та обмеження, які застосовує UI, — з перевіркою на кожній ланці, що падає в разі розбіжностей.

Побудувала всю базову інфраструктуру з нуля

Як одна з перших співробітниць, спроєктувала та побудувала всю базову інфраструктуру та супутні процеси з нуля для SaaS-стартапу в зеленій енергетиці, заклавши фундамент для швидкого зростання.

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 власні моніторинг помилок і трасування

Побудувала власний моніторинг помилок і трасування OpenTelemetry, а не купила готові — очищення payload, виявлення сплесків і регресій, символікацію та синтетичний heartbeat — за 11 операторськими поданнями.

Ваш код на вашому власному сервері

Розгорнула власний git-сервер компанії на Soft Serve, приватний за замовчуванням і без вебпанелі, з портом SSH, прив'язаним до loopback за проміжним хостом (jump host), і зробила цільову сторінку перед ним артефактом збірки основного сайту, а не копією, яку підтримують вручну.

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.

Віддзеркалила сайт у Gemini та Gopher

Побудувала вихідні формати Gemini та Gopher з того самого джерела сайту; запис досягнення фіксує по 777 документів на тому етапі та нуль змінених байтів HTML від додавання форматів.

Виправила мапу сайту з однією позначкою часу

Виправила мапу сайту, де 172 з 176 URL мали одну позначку часу зміни, взявши дату з історії git, після того як з'ясувала, що експорт переписує кожен файл під час кожного запуску.

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.

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.

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.

Задіяла невикористану половину ядра

Задіяла ту половину ядра обміну повідомленнями, якою застосунок ніколи не користувався, — передавання резервної копії, зникомі повідомлення, редагування та повторне надсилання, підтверджені запрошення, проксі та політику шифрування, — проганяючи кожен тест на справжній бібліотеці без заглушок.

Потрібна послуга: DevOps та автоматизація CI/CD?

Оцініть відповідний досвід за пов’язаними роботами. Розкажіть, що потрібно створити; обговоримо обсяг, підхід до виконання та передачу результатів.

Зв'язатися з нами Повне портфоліо

Визначте обсяг до замовлення

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 →

Обговоріть свої вимоги

Розкажіть, що вам потрібно за напрямом «DevOps та автоматизація CI/CD», зокрема про строки та команду, яка підтримуватиме рішення. Обговоримо практичний шлях уперед.

Зв'язатися з нами

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://software.engineer.company/uk/services/devops-cicd/