Engineer

Python

31 件の実績

システムの各部分を結び付ける

「Python」に分類した実績は、稼働するシステムの一部分を示しています。その部分を他の業務につなげるお手伝いをします。 月末に誰かが表計算ファイルをコピーしなくても、決まった時刻に必要な場所へ数値が届く仕組みをつくります。

サービスを見る お問い合わせ →

比較できる課題を探す

ご自身の状況に近い事例をご覧ください。振り返りでは、技術名だけでは分からない、結果を導いた判断を説明しています。 14部門を支える共通インフラ計画 — 14 共通インフラ計画の対象となった研究部門

14部門を支える共通インフラ計画 お問い合わせ →

30日分の電力データを180 msで処理

Python、SQL、Bash + Jqで時間別の電力消費量を集計する複数の実装を構築。Pythonの測定では、30日分のJSONLデータセット1つを180 msで処理した。

ETL/ELTパイプライン6本を設計・運用

Google BigQuery、MSSQL、PostgreSQL、Shellスクリプト、PL/pgSQL、Transact-SQLを用いて6本のETL/ELTパイプラインを設計・実装・管理し、Python APIで効率的に処理できるようデータを統合した。

マニュアル付きPower BIプロジェクト8件を提供

Microsoft Power BIをNodeJS APIとPython FastAPIに連携させ、データ分析と可視化のための8件のPower BIプロジェクトを、詳しいマニュアルとともに納品した。

GitHub ActionsでGIS SaaSのデプロイを自動化

GitHub Actions CI/CD、Python、Bash、SQLで、GIS SaaSアプリケーションのデプロイ、データ処理、レポート作成を自動化した。

Barmanで重要データのバックアップ100件を自動化

Barman、Google Cloud、Bash、Pythonで重要データのバックアップ100件を自動化し、データベース間のデータ整合性を確保した。

30件以上のGIS開発を構築からサポートまで主導

PostgreSQL、Bash、Python、JavaScript、GDAL、ArcGIS、PostGIS、Mapboxの専門知識を活用し、30件を超えるGISプロジェクトの開発・デプロイ・サポートを主導した。

GitHub・GitLab CI/CDで4,000時間を削減

GitHub、GitLab、Bash、PythonによるCI/CDを導入し、データ分析とソフトウェア開発の工程を効率化して4,000時間を削減した。

データ分析レポートシステムを開発

PythonでPDFレポートを作成するデータ分析レポートシステムを開発し、ソフトウェアの四半期売上を400%増加させた。

PythonとPL/pgSQLでデータ処理を自動化

Shellスクリプト、PL/pgSQL、Python、Transact-SQLでデータ処理を自動化し、生産性と効率を向上させた。

HTML/HTML5でWebアプリケーション100個を開発

HTML/HTML5、CSS/SCSS、Django、WordPress、Joomlaを用いて100個のWebアプリケーションを開発し、多様なオンライン展開を支えた。

IPTV用Djangoアプリを構築・保守

IPTVプラットフォームのDjango Webアプリケーションを開発・保守し、新機能の提供と性能・安定性の改善を行った。

20レコードのDNSゾーンをコードで照合

20レコードのDNSゾーンをCloudflare APIと宣言的に照合し、監査とBIND形式の書き出しを別の実行入口として提供。CDNプロキシは構築後、プライバシーの観点から再び無効にした。

2台目のサーバーをコードで構築

別のクラウド事業者で2台目のサーバーの構築を自動化。先にファイアウォールを作成し、起動時から保護されるようにし、両事業者のファイアウォールをREST APIから直接操作して、第三者のcollectionへの依存を避けた。

22静的検査のコミットゲートを構築

HTML、CSS、JavaScript、Python、YAML、Markdown、Shell、リンク、綴り、秘密情報、組版を対象に、1行で説明できる検査22個と詳しい説明が必要な検査5個からなるコミットゲートを構築し、警告扱いや行内抑制を認めない方針にした。

30分も待たずに済む品質検査

最長の検査から順に、最大4並列のワーカープールへ割り当て、ブラウザーを使うサイト品質ゲートを1,636秒から615秒に短縮。名前順では224秒に対して320秒を要するという測定を踏まえた。

SQLiteエクスポートから495ページを生成

読み取り専用のSQLiteエクスポートから3言語495件の実績ページを生成し、ページのアドレスをデータとして定義することで、文章を直してもURLが変わらずリンクが切れないようにした。

更新日時が共通のサイトマップを修正

エクスポートが毎回すべてのファイルを書き直すことを確認し、git履歴から更新日を取得するよう変更。176件のURL中172件が同じ更新日時だったサイトマップを修正した。

文書を独自の静的検査で管理

文書に、引き下げのみ可能なファイル別行数上限、目次要件、実在パス検査を設けた。完了扱いの9計画中6計画に未完了項目があり、コミットを制御する14タスクがどの文書にも記載されていないことを検出した。

10,242行の品質ゲート自体に静的検査を適用

リポジトリ最大のコード群でありながら検査対象外だった品質ゲートのJavaScript 10,242行に整形・静的検査を適用し、13件の指摘を抑制せずに修正した。

データベース駆動の履歴書生成器を構築

19段階の冪等な処理で組み立てる23テーブルのSQLiteデータを単一の情報源として、6形式を出力する、データベース駆動の履歴書・推薦状・実績集・応募書類生成器をPythonの41モジュール、10,580行で構築した。

生成器に981テストと92%カバレッジを適用

警告を失敗扱いにし、分岐カバレッジ92%を下限として生成器に981件のテストを適用。空のファイルから全構築処理を2回実行し、2回目に何も変わらないことを要求して冪等性を検証した。

PDF/UA-1プロファイルの106項目中106項目に合格

9文書のPDF/UA-1検証を記録し、設定済みアクセシビリティプロファイルの検査106項目中106項目に合格した一方、アーカイブ版は146規則のうち1項目で不合格だった。

すべての静的検査ルールをエラーに設定

Python静的検査の全ルールをエラーとして有効にし、1,815件の指摘を0件まで解消。少数の例外には理由を書き、うち2件はコメントだけでなく検査によって根拠を守った。

522行のQR符号器を同梱

実行時の3つ目の依存関係を追加せず、GF(256)上のReed-Solomon、固定モジュール配置、8つのマスクパターンを持つQR符号器を522行で同梱し、独立に実装した復号器で完成行列を読み戻して検証した。

検査自体のテストを作成

正常入力しか与えない検査は、何も読まずに正常と報告するようになり得ると確認し、検査自体のテストを作成。誤字を埋めて綴り検査の検出を確かめ、テスト内の固定値ではなくデータベースからID範囲を取得した。

フックと品質ゲートの不一致を発見

同じ検査を行うという文書の記述に反し、コミットフックと品質ゲートが異なる検査を行っていたことを、一覧を比較するテストで発見。手動でしか動かなかった5個は履歴書の文章を検査するものだった。

gitホスティングCIを試し2不具合を発見

gitホスティング側のCIフックを予行実行し、ファイルを読むだけでは到達できない2件の不具合を発見。解決不能な引数を入力へ渡すフォールバックと、検査先に引き継がれ19テストを失敗させるgit自身の環境変数を修正した。

すべての文字列をコンテンツツリーへ移行

実際には使われていないと気づけない翻訳と、実績の3分の1だけを黙って評価する検査を発見し、利用者向け文字列をすべてPythonから3言語874ファイルのコンテンツツリーへ移した。

言語間の数値照合を構築

英語原文にある数値の欠落と、対象言語に合わない表記で失敗する言語間コンテンツ検査を構築し、指標が抜けたデンマーク語説明2件と英語式引用符を使うウクライナ語16箇所を検出した。

追跡せずに13アドレスを測定

Cookie、トラッカー、第三者サービスを使わず、Webサーバーのアクセスログのみから13アドレスを測定。書き込み時に接続元ネットワークをマスクし、21の日別集計と20の月別分析軸からなる永続アーカイブへまとめた。

6収集処理が省略した入力を明示

6つのレポート収集処理が、未読ファイル、未導入の検査、認識できなかったjournal行など、読み取れなかった入力範囲を明示するよう変更。収集不能と正常なホストが同じページになると確認し、116件の実測サンプルから閾値を定めた。

技術的な取り組みを絞り込む

代表的なデータセット、そこから答えを得たい問い、現在のボトルネックをお知らせください。品質、処理量、復旧の測り方を合意できます。

料金を見る · お問い合わせ →

自社の状況に当てはめる

お使いのシステムで「Python」がどのように関わっているかをお聞かせください。適したサービスを選び、その業務範囲を定めるお手伝いをします。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/categories/python/