Engineer

Technologies

Les technologies d'Engineer ApS : langages, plateformes de données, cloud et infrastructure, et notre façon de décider lesquelles utiliser.

Un outil est un moyen. Ce qu'un client achète, c'est le discernement qui permet de choisir le bon et la discipline pour bien l'exploiter — mais vous avez demandé avec quoi nous travaillons, le voici donc. Chaque ligne est étayée par des réalisations dans le portfolio.

Nous écrivons dans **les langages que votre prochaine recrue connaît déjà** — pour que le code ne soit jamais le nôtre uniquement.



Python Go TypeScript JavaScript PHP

Adaptez les outils aux travaux

Choisissez les outils selon les travaux que votre équipe doit effectuer et le système qu'elle maintiendra.

Découvrir nos prestations Contactez-nous →

Examinez les décisions technologiques

Explorez les réalisations qui sous-tendent un choix technologique et demandez pourquoi il répondait à ces contraintes précises.

Études de cas Contactez-nous →

Langages et environnements d'exécution

Python pour les données, l'automatisation et les services. **Go** lorsqu'un binaire statique unique et une faible consommation de ressources comptent. **TypeScript** et JavaScript moderne pour la couche Web. **SQL** comme langage à part entière

plutôt que comme résultat généré par un ORM — la plupart des travaux de performance pour lesquels on nous sollicite se situent là. **Bash** pour relier l'ensemble, écrit pour être lu.

Données

PostgreSQL est notre premier choix, et nous l'exploitons largement avant de chercher autre chose — partitionnement, réplication, extensions et planificateur de requêtes. Nous exploitons aussi **MS SQL Server**, **BigQuery**, **SQLite** et **Redis**, ainsi que les outils spatiaux qui s'y ajoutent : **PostGIS**, **QGIS**, **GDAL** et **ArcGIS**. Les pipelines de traitement sont conçus pour être réexécutables et idempotents, car après un incident, la question est toujours « peut-on le rejouer ? ».

Des outils choisis **pour le travail à réaliser** — avec une explication des compromis.

Le choix dépend de vos systèmes existants, de vos contraintes et du travail à réaliser.

Cloud et infrastructure

Google Cloud et **AWS**, avec **Azure** lorsque le client y est déjà installé. **Linux** sous l'ensemble — Debian et Ubuntu. **Docker** et **Kubernetes** pour les charges qui les justifient, de simples unités systemd pour les autres. **Ansible** et **Terraform** décrivent les machines ; un environnement est ainsi un document plutôt qu'un chantier archéologique. **Caddy** et **nginx** en façade.

Exploitation en production

Prometheus et **Grafana** pour les métriques, une journalisation structurée pour le reste et des alertes déclenchées par des symptômes perceptibles par un utilisateur plutôt que par chaque soubresaut d'une courbe. Des tâches CI qui compilent, testent, analysent et déploient sans intervention humaine. Automatisation des sauvegardes et procédures distinctes de test de restauration.

La couche Web

Le statique d'abord lorsque cela convient, car une page statique peut être servie sans couche applicative exécutée au moment de la requête : **Hugo** construit ce site. **Django** et **PHP** lorsqu'il y a une application à servir. Du HTML sémantique, du CSS écrit à la main et une accessibilité traitée comme une exigence plutôt qu'un constat d'audit.

Notre façon de choisir

Une technologie éprouvée choisie délibérément vaut mieux qu'une technologie nouvelle choisie par automatisme. Nous choisissons selon le problème, ce que votre équipe pourra exploiter après notre départ et la manière dont le système échoue plutôt que celle dont il se présente en démonstration. Lorsque la bonne réponse est l'ensemble de technologies dont vous disposez déjà, nous le disons — et sinon, nous le disons aussi, avec un raisonnement écrit.

Ce site est son propre exemple concret : la page des crédits nomme chacune de ses technologies, jusqu'à la police et à l'outil de sauvegarde.

Si vous préférez un fournisseur avec des convictions plutôt qu'une liste de préférences, dites-nous ce que vous construisez.

Nous travaillons avec **les plateformes** sur lesquelles votre entreprise s'appuie déjà.

Apple Microsoft DigitalOcean AWS Google

Découvrir nos prestations

Côté serveur, nous utilisons **Linux en logiciel libre** — sans licence à payer ni rien de caché.



Ubuntu



Debian

Comment nous exploitons les serveurs

Une liste d'outils **n'est pas une qualification** — savoir lequel répond à votre problème en est une.

Chaque technologie citée sur cette page étaye des réalisations que vous pouvez consulter et lire.

Voir leur utilisation dans les réalisations Nos prestations →

Partez des technologies dont vous disposez

Présentez vos technologies existantes, vos exigences opérationnelles et les compétences disponibles dans votre équipe.

Consulter les tarifs Contactez-nous →

Discutez d'un changement technique justifié

Discutez du plus petit changement utile avant de vous engager sur une nouvelle plateforme ou un nouveau framework.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark



Emportez cette page avec vous



Fiche de contact

Créé avec soin et amour par D.

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

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/tech-stack/