

Coût de développement

Claude Code · 13 plateformes · 6 → 23 août 2026

8650\$_{US}

équivalent API estimé — 314 890 lignes de code livrées, soit $\approx 27 \$ / 1000$ lignes



Lou-Ka
www.lou-ka.com



Groupe KA
www.groupe-ka.com



Immo-Ka
www.immo-ka.com



Vrai-Prix
www.vrai-prix.com



Job-Ka
www.job-ka.com



Fabri-Ka
www.fabri-ka.com



Auto-Ka
www.auto-ka.com



Sorti-Ka
www.sorti-ka.com



Trouve-Ka
www.trouve-ka.com



Créa-Ka
www.crea-ka.com



Resto-Ka
www.resto-ka.com



Food-Ka
www.food-ka.com



API-Ka
www.api-ka.com

Synthèse

Entre le 6 et le 23 août 2026, l'ensemble de l'écosystème Groupe KA — 13 plateformes web, leur infrastructure et leur outillage — a été développé avec ClaudeCode. L'analyse de 348 sessions (laptop + 5 nœuds du cluster MacLustr) chiffre ce développement à **8 650 \$ US en équivalent API** (fourchette : 7 440 \$ en attribution stricte), pour **314 890 lignes de code** en production.

Coût total estimé

8 650 \$

attribution stricte : 7 440 \$

Lignes de code

314 890

1 824 fichiers · 13 sites

Sessions Claude Code

348

28 600 réponses du modèle

Tokens générés

17,8M

+ 4,4 G tokens lus en cache

Durée

18 jours

6 → 23 août 2026

Efficiencia globale

≈ 27 \$

par 1 000 lignes livrées

Où le travail s'est fait

répartition du coût par lieu d'exécution des sessions



Sessions «home» (remote-first) 4 686 \$ · 54,2 % **Dossiers-projets Ka (laptop)** 3 287 \$ · 38,0 %

Sessions sur les nœuds 678 \$ · 7,8 %

Modèles utilisés

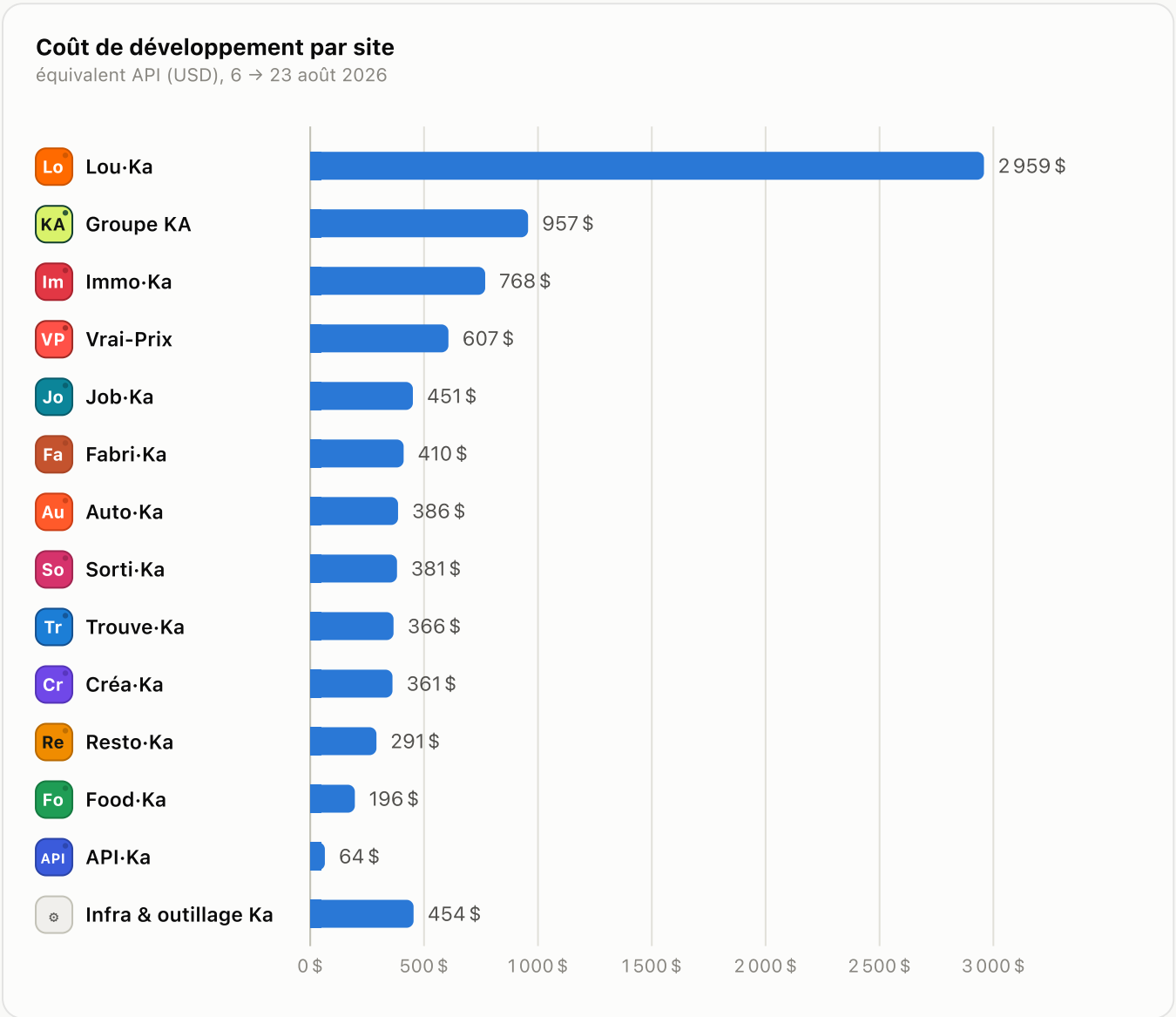
part du coût par famille de modèle

Claude Fable 5 7 761 \$ · 89,7 % **Claude Opus 4.8 / 5** 886 \$ · 10,2 %

Haiku / Sonnet < 3 \$ · < 0,1 %

Coût Claude par site

Sessions dédiées à chaque site (dossiers-projets du laptop et des nœuds), plus les sessions génériques attribuées à leur application dominante. Lou-Ka concentre à elle seule **34 %** du coût — cohérent avec son poids dans le code (26 % des lignes) et ses 286 connecteurs Python.



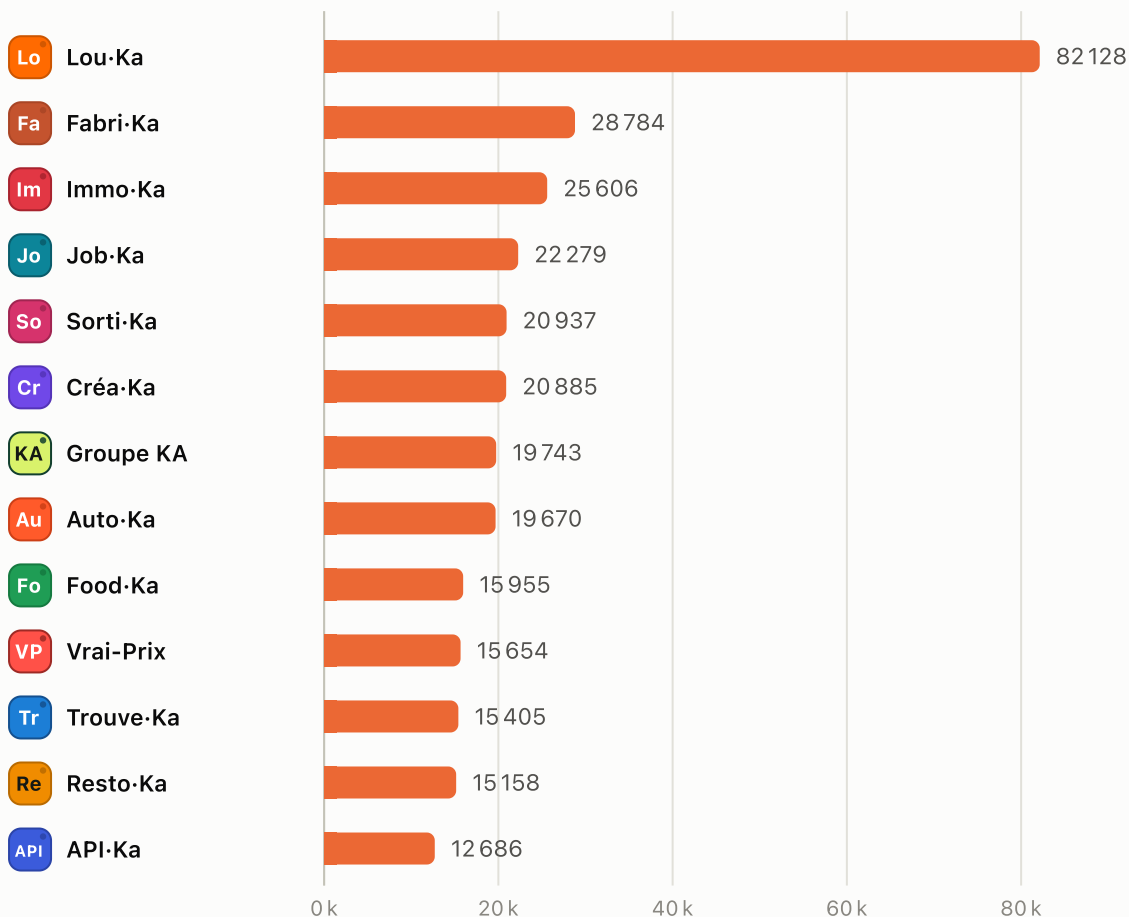
« Infra & outillage Ka » regroupe ka-orchestrator (115 \$), ka-bot (126 \$) et les projets Ka annexes — forma-ka, ora-ka, toit-ka, poster-ka (213 \$).

Le code livré

Comptage sur les repos git des nœuds de déploiement (source de vérité du développement remote-first) — fichiers de code uniquement, sans dépendances, builds ni fixtures de test.

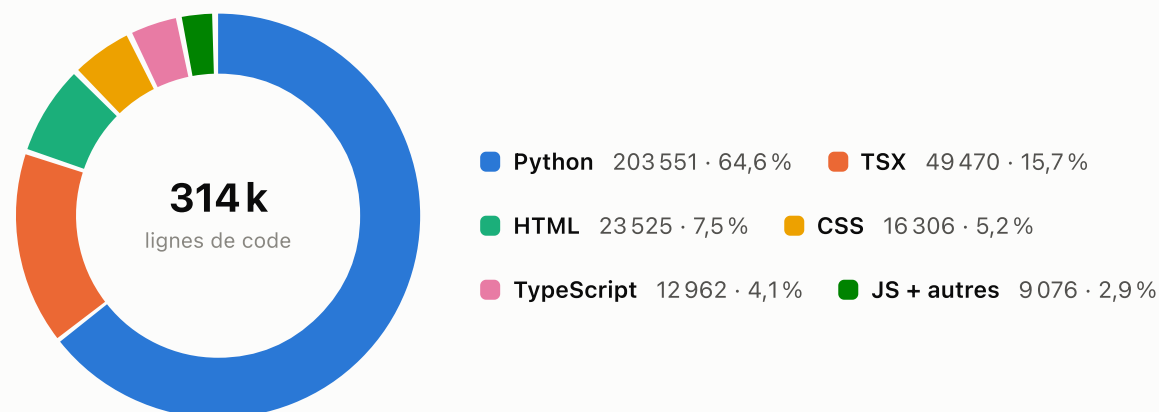
Lignes de code par site

git ls-files · py, ts, tsx, js, css, html, sh, sql



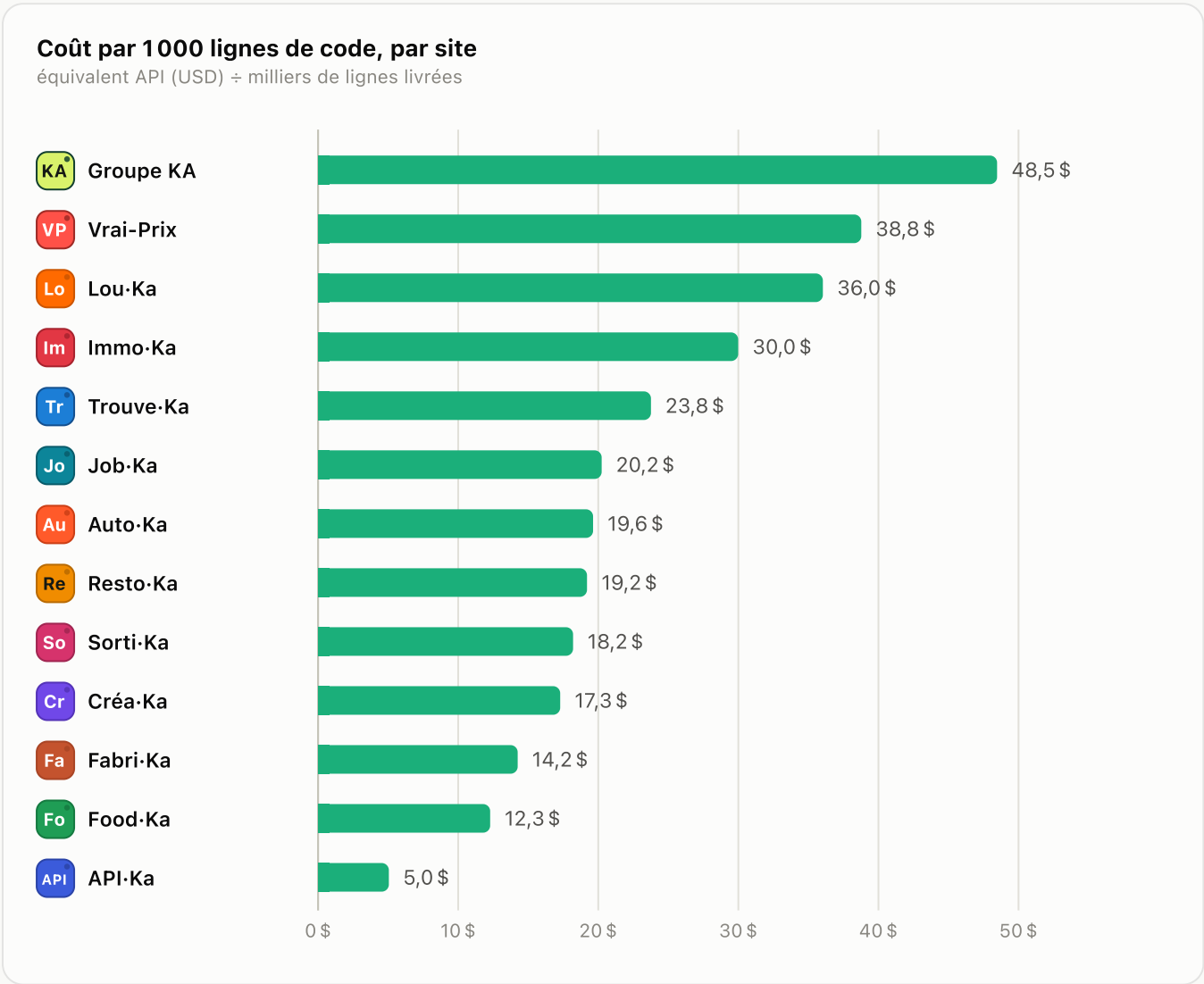
Répartition par langage

Python domine : connecteurs, moteurs de qualité et API



Efficiency — coût par millier de lignes

Le ratio coût/code varie d'un facteur 10 selon la nature du travail : le hub Groupe KA porte le design system, le SSO KAID et beaucoup d'infrastructure partagée pour peu de lignes dans son propre repo ; à l'inverse, API-Ka et Food-Ka sont surtout du backend généré efficacement.



Moyenne des 13 sites : 26,0 \$ / 1000 lignes (27,5 \$ en incluant l'infra). Les sites riches en analyse, cartographie et validation (Vrai-Prix, Lou-Ka, Immo-Ka) coûtent naturellement plus par ligne.

Détail par site

Vue d'ensemble — code, déploiement et coût

classement par coût de développement décroissant

SITE	DOMAINE	NŒUD	LIGNES	FICHIERS	COÛT CLAUDE	\$ / 1 000 LIGNES
 Lou·Ka	lou-ka.com	M3U96b	82 128	370	2 959 \$	36,0 \$
 Groupe KA	groupe-ka.com	M3U96b	19 743	93	957 \$	48,5 \$
 Immo·Ka	immo-ka.com	M4M64a	25 606	104	768 \$	30,0 \$
 Vrai-Prix	vrai-prix.com	M3U96a	15 654	93	607 \$	38,8 \$
 Job·Ka	job-ka.com	M3U96a	22 279	423	451 \$	20,2 \$
 Fabri·Ka	fabri-ka.com	M4M64a	28 784	99	410 \$	14,2 \$
 Auto·Ka	auto-ka.com	M4M64b	19 670	76	386 \$	19,6 \$
 Sorti·Ka	sorti-ka.com	M3U96a	20 937	92	381 \$	18,2 \$
 Trouve·Ka	trouve-ka.com	M2M32	15 405	134	366 \$	23,8 \$
 Créa·Ka	crea-ka.com	M3U96b	20 885	100	361 \$	17,3 \$
 Resto·Ka	resto-ka.com	M3U96b	15 158	64	291 \$	19,2 \$
 Food·Ka	food-ka.com	M4M64b	15 955	110	196 \$	12,3 \$
 API·Ka	api-ka.com	M3U96b	12 686	66	64 \$	5,0 \$
Total 13 sites			314 890	1 824	8 197 \$	26,0 \$
+ Infra & outillage Ka (ka-orchestrator, ka-bot, projets annexes)					454 \$	

Méthodologie & mises en garde

Méthode

1. **Sources.** Transcripts JSONL de Claude Code (`~/ .claude/projects`) sur le laptop et les 5 nœuds du cluster exécutant Claude Code (M3U96a, M3U96b, M4M64a, M4M64b, M2M32) — 348 sessions analysées, avec déduplication des messages par identifiant.
2. **Tarification.** Tarifs API Anthropic officiels : Fable 5 à 10 \$ / 50 \$ par MTok (entrée/sortie), Opus à 5 \$ / 25 \$, écriture de cache à 1,25× l'entrée, lecture de cache à 0,1×.
3. **Attribution.** Sessions des dossiers-projets Ka attribuées directement ; sessions génériques (travail remote-first via SSH) attribuées au site dominant par analyse des mentions dans le transcript.
4. **Comptage du code.** `git ls-files` sur les repos des nœuds de déploiement, extensions de code uniquement ; 5,9M de lignes de fixtures HTML de test (Lou·Ka) exclues.

Mises en garde

1. **Équivalent API, pas une facture.** Sous abonnement Claude (Max), le coût réellement payé est celui de l'abonnement — ces montants mesurent la valeur API du travail effectué.
2. **Fenêtre de rétention.** Claude Code purge les transcripts après ~30 jours : le rapport couvre le 6 → 23 août 2026. Le développement Ka est concentré sur cette période, mais tout travail antérieur (ex. genèse de Vrai-Prix) n'est pas compté — le total est donc un plancher.
3. **Attribution approchée.** Les campagnes transverses (mobile 13 sites, stats v2, KA Agent...) sont attribuées à un seul site dominant ; les totaux globaux, eux, sont exacts.
4. **Cache.** Écritures de cache facturées au taux 5 min (1,25×) ; si du cache 1 h a été utilisé, ajouter environ +5 à +8 %.

Groupe KA — Simon-Pierre Boucher · rapport produit par Claude Code (Fable 5) · 23 août 2026