

Степан Коток (Stepan Kotok)

Tech Lead та AI Product Builder, Solution Architect

Випуск продуктів зі ШІ-агентами · Програмна архітектура · Керівництво командою · Claude Code, MCP, LLM

Англійська – B2; українська, російська – рідні.

Кишинів, Молдова · [+373 78 722 382](tel:+37378722382) · wa.me/37378722382 · t.me/stkotok · [Viber](https://viber.com/uk/stkotok) · stkotok@gmail.com · linkedin.com/in/stkotok · github.com/StKotok · stepankotok.com



Профіль

Швидко перетворюю продуктову ціль на робоче, перевірене ПЗ. Код пишуть ШІ-агенти, а я відповідаю за продуктове мислення, архітектуру, критерії приймання, безпеку та реліз і читаю кожен diff.

Роль у команді

- Основна роль – техлід, який відповідає за надійну роботу продукту: обираю технології, проєктую архітектуру, керую ШІ-агентами, що пишуть код, і контролюю якість на всіх етапах.
- Керую й командами: у [Neatapps](#) координував роботу із замовниками та командою, проводив щоденні стендапи зі звітом про виконану роботу та розбором перешкод.

Як я пишу код зі ШІ

Інструменти

Claude Code – основний інструмент, на найвищому тарифі Max. Субагенти, навички, хуки, власні MCP-сервери, паралельна робота в ізольованих git worktree. Модель добираю під завдання: потужніша планує й приймає роботу, швидша пише код за завданням із критеріями приймання.

ChatGPT/Codex, Gemini, Grok – на платних тарифах Plus, Pro і Super: незалежне рев'ю коду та планів. Кілька моделей по черзі, з виправленнями між раундами; друга думка – перед незворотними кроками (міграцією, розгортанням, зміною прав доступу).

Перевірка в браузері та на телефоні (Playwright, Mobile-MCP) – агенти самі перевіряють результат у справжньому браузері та на мобільному телефоні: наскрізні сценарії, доступність, знімки екрана.

Як це пришвидшує роботу, не знижуючи якості

- Швидше код.** За чотири дні зібрав зі ШІ-агентами платформу онлайн-курсів з офлайн-режимом, входом і правами доступу та панеллю адміністратора, покрити наскрізними автотестами.
- Швидше й ґрунтовніше тестування.** Агенти пишуть автотести в коді та сценарії для ручного тестування. На проєкті [ШІ-асистента для туристичного сервісу](#) (нижче) за сім днів ми з агентами додали 1 310 тестів у шести шарах. Також ШІ тестує, як живий тестувальник: проходить сценарії в справжньому браузері та на мобільному телефоні, робить знімки екрана й повідомляє, що виправити.
- Швидші й безпечніші релізи.** Блокувальна перевірка перед основною гілкою, кожна зміна пов'язана із завданням, CI в GitHub Actions.
- Перевіряю, а не вірю на слово.** Якщо ШІ помилився, фіксую для нього правило, щоб помилка не повторилася. Будь-який згенерований код спочатку перевіряю типізацією, тестами та архітектурним рев'ю і лише потім приймаю.

Вибрані проєкти

ШІ-асистент для туристичного сервісу: готелі, маршрути, бронювання

2026 · Next.js, TypeScript, Supabase/PostgreSQL, Claude API та OpenAI API, Vitest, Playwright, GitHub Actions

Проект Willy Agency, Саас-Фе, Швейцарія · [Туристичний сервіс](#) · триває: злито 155 pull request'ів станом на 24.10.2026

Власник створив продукт зі ШІ-агентами за 71 день; продукт працював і був близький до перших продажів. Першого тижня, у серпні 2026 року, керуючи ШІ-агентами, я:

- Зробив надійною роботу діалогового асистента, що спирається на базу знань туристичного сервісу (готелі, маршрути, сервісні запитання гостей, бронювання слотів).

- Додав з нуля 1 310 тестів у шести шарах, без мережі, бази даних і моделі; повний прогін займає 3,64 секунди й не потребує секретів.
- Провів аудит безпеки: авторизація маршрутів, ізоляція даних на рівні рядків (RLS у Supabase, задумана, щоб модель не дісталася до приватних даних при prompt injection), вхідні вебхуки, секрети, розділення середовищ; критичне закриття і закріпив тестами, через які збірка падає, якщо проблема повертається.

Doodlemigo – застосунок для малювання для дітей 3–7 років

2026 • власний проєкт • зроблено зі ШІ-агентами

За п’ять годин власного часу зібрав зі ШІ-агентами застосунок для малювання, включно з добором графіки та продумуванням функцій: обвести картинку пальцем, розфарбувати й побачити, як вона оживає; працює офлайн, малюнком можна поділитися за посиланням.

Застосунок для оцінки стану пристроїв

Грудень 2022 – грудень 2025 • Flutter, нативні модулі iOS і Android, Node.js, Angular, TensorFlow Lite, Core ML

Neatapps • керівник проєкту й архітектор рішення • під NDA, міжнародна компанія електроніки

- Провів проєкт від концепції замовника до релізу: вимоги, обсяг робіт, беклог, планування.
- Зібрав і очолив команду: найняв дизайнерів і розробників, яких бракувало, налаштував процес за Scrum.
- Керував розробкою комп’ютерного зору на пристрої: виявлення пристрою в кадрі в реальному часі, автоматична зйомка, модель розпізнавання відблисків.

Досвід роботи

Willy Agency – технічний партнер (за контрактом)

Серпень 2026 – дотепер • неповна зайнятість • Саас-Фе, Швейцарія, віддалено

AI Product Builder і технічний лід (за контрактом)

Квітень 2026 – дотепер • Кишинів

- Доводжу до продакшену прототипи, зібрані за допомогою ШІ: архітектура, дані, авторизація, платежі, тести критичних сценаріїв, реліз.

Neatapps – співзасновник, технічний керівник проєктів

Липень 2020 – березень 2026 • замовники зі США та Європи

- Перетворював цілі замовників на технічний обсяг робіт: дослідження, вимоги, оцінки, план поставки, спілкування із замовником.
- Координував розробників і дизайнерів на всьому циклі: архітектура, розробка, тестування, реліз, підтримка після релізу.

Neatapps – співзасновник, мобільний розробник

Липень 2015 – липень 2020

- Розробка власноруч: Java, Android, iOS (Swift).

Навички

Продукт і поставка:	перетворення цілі на обсяг робіт і критерії приймання, беклог і планування, оцінки, релізи, спілкування із замовником.
Керівництво:	найм і керівництво розробниками та дизайнерами, Scrum, рев’ю коду й планів, відповідальність за технічні рішення й ризики.
Архітектура й якість:	архітектура рішень, безпека даних і доступу (безпека на рівні рядків, авторизація, секрети), стратегія тестування, CI/CD, плани релізу та відкату.
ШІ-агенти та LLM:	Claude Code (субагенти, навички, хуки, паралельні worktree), Codex, Cursor, Gemini; власні MCP-сервери; LLM-функції з викликом інструментів і RAG; рев’ю планів і коду кількома моделями.
На чому працюють мої продукти:	TypeScript, Next.js, React, Node.js, PostgreSQL/Supabase, Cloudflare Workers, Vercel, Playwright, Vitest, GitHub Actions – код пишуть ШІ-агенти під моїм рев’ю.