

DevEx Chat + Industry Q&A
University of Puerto Rico, Río Piedras
Raúl Negrón-Otero - September, 2026

Motivation

- I wanna hear from you!
- You might find DevEx interesting
- It's OK if we don't use up all the time

Who Am I? 🙋

- Raúl Negrón-Otero
- <https://raulnegrón.me>
- Clase de CCOM UPRRP - 2018
- Nintendo mega fan
- Senior Software Engineer (Fullstack)
 - Abartys Health (PR startup)
 - Alias Payments (PR startup)
 - BrainHi (PR startup)
 - Zapier (remote)
 - Freelancer (local and remote)
 - PostHog (remote)

What We'll Talk About

1. How do web development companies ship software today?
2. What is DevEx, and why do companies hire for this role?
3. Q&A!

How do companies ship software today?

- It depends...

How do companies ship software today?

- It depends...
- In web development, there is one approach that is common for backend systems: **containerization**

How do companies ship software today?

- It depends...
- In web development, there is one approach that is common for backend systems: **containerization**
- Frontend systems very likely leverage cloud storage + CDN (or Vercel...)

**How does a container (or frontend bundle)
get built and shipped in the first place?**



Code goes in... things happen... and a release bundle/container image comes out! How?

- Teams use Git repositories to manage codebases
- Code is merged into a specific Git branch (like `main`)
- Continuous Integration (CI) enabling systems like GitHub Actions build and push the container image

A simplified code-to-release pipeline ("golden-path")

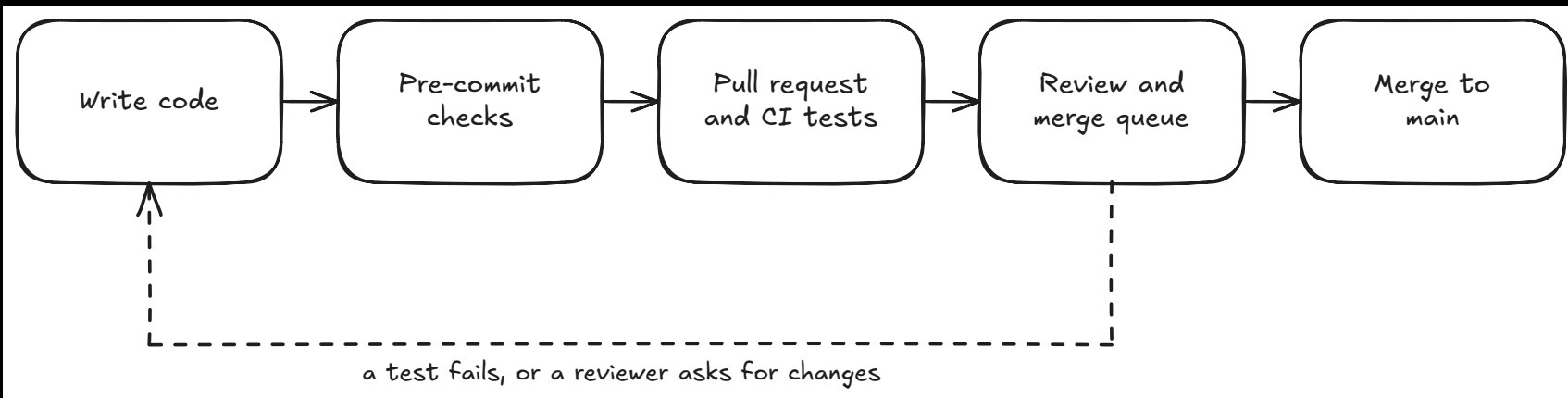
1. Developer writes code

A simplified code-to-release pipeline ("golden-path")

1. ~~Developer writes code~~ Developer prompts their LLM

A simplified code-to-release pipeline ("golden-path")

1. ~~Developer writes code~~ Developer prompts their LLM
2. Pre-commit checks + linting
3. git commit + Pull Request created
4. CI system runs tests + more
5. Human or bot reviews and then approves / rejects PR
6. PR placed in the merge queue
7. Merge commit to main!



An engineer interfaces with a lot of systems day-to-day

- Terminal or IDE (user configurable)
- LLM Agent (user configurable)
- Programming language (user configurable)
- Tests + other developers' code
- Git + pre-commit checks
- Pull Request + Code Review
- Continuous Integration workflows
- and more...

An engineer interfaces with a lot of systems day-to-day

- Terminal or IDE (user configurable)
- LLM Agent (user configurable)
- Programming language (user configurable)
- Tests + other developers' code
- Git + pre-commit checks
- Pull Request / Merge Request + review
- Continuous Integration workflows
- and more...

 **DevEx owns these!**

**So, what are Developer
Experience engineers?**

So, what are Developer Experience engineers?

- Engineers that own the development tooling and processes
- Work at the cutting edge of agentic development / steering
- Bottom line: we help other devs work better

Tools used at PostHog **(<https://github.com/posthog/posthog>)**

- **Code Repo:** GitHub
- **Container Image Repo:** Docker Hub + AWS ECR
- **Local Toolchain Manager:** Flox
- **CI/CD:** GitHub Actions + Depot
- **Merge Queue:** Trunk
- **Remote Devboxes:** Coder
- **Automated AI code review:** Greptile
- **LLMs:** Mostly Claude Code
- **Programming Languages:** Python (Django), JavaScript (Node.js / React) Rust, Go, Bash, etc...

Highlighting some of my DevEx efforts at PostHog...

feat(hogli): add ci:preflight pre-push hook, checks, and telemetry #65581 

Code ▾

 Merged [rnegron](#) merged 20 commits into [master](#) from [worktree-ci-preflight-v0](#)  on Jul 2

 Conversation 7

 Commits 20

 Checks 197

 Files changed 15

+945 -179 

Highlighting some of my DevEx efforts at PostHog...

feat(engineering_analytics): CI analytics scene #61182 

Code 

 Merged [rnegron](#) merged 45 commits into [master](#) from [rauln/eng-analytics-ci-scene](#)  on Jun 15

 Conversation 17

 Commits 45

 Checks 244

 Files changed 51

+9,595 -83 

health, pull requests, workflows, and cost for the connected repo.

- Overview
- Pull requests
- Workflows
- Teams
- Test health
- Health

posthog

PostHog/posthog · [View on GitHub](#)

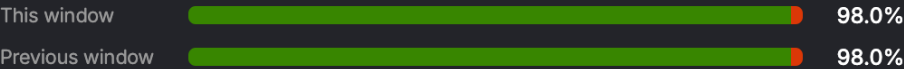
PostHog/posthog

Last 7 days

CI health

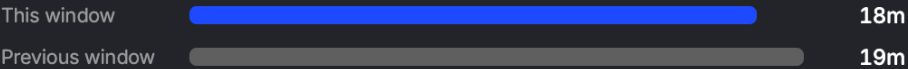
CI pass rate

98.0% ±0pt



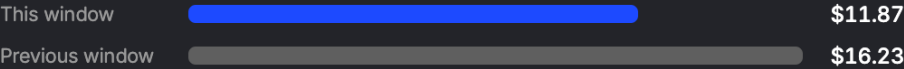
Median time from push to all checks green

18m ▼ 8%



CI cost per merged PR

\$11.87 ▼ 27%



Master health

CI duration per commit

recent 48 commits



Top workflows by cost

\$13.7k total CI spend

A look at PostHog's Q3 DevEx roadmap

- <https://posthog.com/teams/developer-experience>

Goals

Q3 2026 Objectives

North star: make shipping 10,000 PRs a month easy. Agents will write most of the code, so the quarter is about hardening the loop that makes that volume work: write → verify → review/merge → measure.

Write: local dev loop (Raúl Negrón-Otero)

Shorten the inner loop and let agents run the stack and tests locally, with devboxes as a favored way to spin up a full stack. Local agent steering (skills, contexts) is part of this.

Verify: CI loop we trust (Julian Bez , Georges-Antoine Assi , Raúl Negrón-Otero)

Fast, cheap, low-flake CI that people actually believe. Keep sharding optimal against a wall-clock target, move preview and end-to-end onto sandboxes, and drive flakiness down.

Review & merge: agents gating agents (everyone)

As agents author most PRs, no one reviews every diff by hand. Route PRs by blast radius, let independent reviewer agents gate rather than just comment, and lean on deploy and rollback as the safety valve.

Measure: how do we know it's getting better? (everyone)

Every loop above gets a metric: PR-to-merge time, flakiness, CI run times, and per-product attribution so teams own their own cost. Non-goal: measuring individual engineering productivity.

Questions, please 😊

Thanks! See you tomorrow?



FSN
Speakers



- luma.com/fsn
- <https://discord.gg/HagJYKbrWb>