Master Your Local Development Environment for AI adoption

YEONG SHENG 

STANLY 

The Lake and Rocks Metaphor (LeSS - Queueing Theory)

Press Space for next page →

Unlock the full potential of AI integration in your development workflow.
Achieve a tight, full-cycle feedback loop, with rapid iteration.

"Works on my machine"

Isn’t funny when it blocks the entire team/s

In multi-team setup, simple problems become massive obstacles

- Delayed integration
- Hours lost to environment issues
- Developers, new and existing piling code, yet unable to verify changes thoroughly in local enironment
- LeSS exposes these problems like dropping water levels to reveal rocks in a lake
- Prevent teams from using AI tools like Cursor effectively.
- Teams struggle to quickly validate AI-generated code, limiting its potential impact.

{ DEMO }

lunchbox_api

Is it even possible to take complete control of your local environment?

- databases, messaging, and dependencies?
- repeatable, composable, clean-slate DB migrations?
- time-travel testing, and bending your software product system to your will for effective testing & verification?

Learning Outcomes

- **Spot the real damage from broken local environments.** It's not just annoying; it's expensive and blocks both human and AI productivity.
- **Take total control of your local setup:** databases, messaging, dependencies, and runtime in order to verify code changes instantly.
- **Learn advanced testing techniques** including clean-slate resets, time-travel debugging, and the ability to short-circuit any flow for testing.
- **Actually use AI tools like Cursor productively** because you'll have the local verification setup to check if the generated code actually works.
- **Create an improvement roadmap** to start your teams' journey toward better integration practices and effective AI tool adoption.

{ DEMO }

NixOS Nix Package Manager

Jetify Devbox

```
{
  "packages": [
    "postgresql@17.5",
    "beam27Packages.elixir@1.18.4",
    "nodejs@24.3.0"
  ],
  "shell": {
    "init_hook": [
      "echo 'Welcome to devbox for lunchbox_api!' > /dev/null",
      "corepack prepare pnpm@10.13.1 --activate",
      "corepack use pnpm@10.13.1",
      "cd assets && pnpm --frozen-lockfile recursive install"
    ],
    "scripts": {
      "test": [
        "BASIC_AUTH_USERNAME=specialUserName BASIC_AUTH_PASSWORD=superSecretPassword mix test"
      ],
      "setup-db": [
        "devbox services up -b",
        "psql -U postgres -c \"CREATE USER postgres;\" || true",
        "psql -U postgres -c \"CREATE DATABASE lunchbox_api_dev OWNER postgres;\" || true",
        "psql -U postgres -c \"CREATE DATABASE lunchbox_api_test OWNER postgres;\" || true",
      ]
```

The Real Cost of Broken Local Environments

- Integration delays ripple across teams.
- Hours lost to “works on my machine” issues.
- AI tools become unreliable without local verification.
- Blocked productivity: both human and AI.
- AI-generated code is only as good as your ability to test it locally.
- Without a clean, reproducible environment, you can’t trust AI suggestions.
- **Example:** Cursor and similar tools need fast feedback loops.

Learn More

Documentation · GitHub · Showcases

Powered by Slidev