



Better Magento Code, Written with AI

Andreas von Studnitz

Senior Architect



What developers say

AI slop is real

A horrible, horrible idea from a maintenance standpoint.

LT **Linus Torvalds** Creator of Linux, The Register, Nov 2025

A MASSIVE time waster for reviewers – especially if people don't disclose it.

AJ **Adriaan de Jongh** Game developer, via GamingOnLinux, Feb 2026

AI slop PRs are becoming increasingly draining and demoralizing for Godot maintainers.

RV **Rémi Verschelde** Godot maintainer, Bluesky, Feb 2026



The question

So AI makes code worse?

It can. It doesn't have to.



About me

Andreas von Studnitz

- Magento developer since 2008
- Co-founder of integer_net
- Senior Architect at Hyvä since 2024
- Consultant, trainer, speaker
- Active in the Magento community



Hyvä

| 'hyvæ or huu-vah | Finnish ⇌ English

Good, desirable, trustworthy

- Started in 2020 as an alternative frontend
- Today a software vendor improving the whole Magento platform
- Theme, Checkout, CMS, ...
- Focus on performance and developer experience
- Almost 30 people all over Europe
- Rooted in the Magento community

Hue-va

Hoova



Highvaaa

Hyvä

!?.

Background

Hyvä and AI

Not early adopters.

All-in since early 2026.

Claude Code is now part of our daily work:

- Development
- Code reviews
- Documentation



Background

Me and AI

It's complicated.

I don't like what it does to society, environment and culture.

I enjoy using it for development.



Example

Hyvä Portal: case study form

The image displays a web application interface for a case study form. On the left is a dark sidebar with a navigation menu. The main area contains six sequential steps, each in a light gray modal box.

Sidebar Navigation:

- Hyvä—Site
- Pages
 - Previous Rounds
 - Hyvä Portal
 - License Renewal Modal
 - Portal - Phase 2
 - Portal - Phase 3
 - Portal - Additional August 20...
 - Portal - Payment + POS
 - Case Study Directory
 - Universal Add-to-Cart
- Agents
- Implemented ✓
 - Homepage 1.2
 - Homepage 1.2 Mobile
 - Homepage 1.2 Desktop
 - PDP Revamp
 - DHD — Theme

Modal - Case Study - Step 1: Add case study link

Modal - Case Study - Step 2: Merchant details

Modal - Case Study - Step 3: Project details

Modal - Case Study - Step 4: Case study details

Modal - Case Study - Step 5: Review and submit

Modal - Case Study - Step 6: Submitted for review

From Figma to finished workflow, built on existing components.



Example

Hyvä Advanced Admin Permissions

Fine-grained admin roles:

- Multiple roles per user
- Deny rules
- Scopes
- Expiry
- Audit log



Example

Hyvä Advanced Admin Permissions

Base module: 3 days.

Testing, review, compatibility, optimisation, edge cases, docs: 3 months.

Writing code got cheap. Shipping it didn't.



hyvää

1. Why quality can go up

If you do it right



2025 vs. 2026

Agents went mainstream

Developers using AI agents: 31% in 2025, 59% in April 2026.

Developers saying AI writes half their code or more: 12% in 2025, 42% in 2026.

— Stack Overflow survey 2025 and pulse survey 2026 · BairesDev Dev Barometer



Plot twist

AI slop got better

Over the last few months, we have stopped getting AI slop security reports in the curl project.

DS

Daniel Stenberg curl founder, via The Register, Apr 2026



Survey

Developers trust AI more

“The share of AI-generated code reaching production without a separate manual review step jumped from just 7% in early 2026 to 36.3% by mid-May of the same year”

— Spring 2026 developer habits report, Cursor



2025 vs. 2026

Trust comes from Experience

2025: only 33% trusted the accuracy of AI tools.

2026: 87% trust AI for at least part of their work.

The more experience, the more trust: 69% of developers with 16+ years view AI favorably, vs. 53% of juniors.

— Stack Overflow Developer Survey 2025 and 2026



2025 vs. 2026

Developers won't go back

"Previously, we found a 20% slowdown. That finding is now outdated."

— METR, AI research nonprofit · Feb 2026



Magento

Magento suits AI

- Strict structure
- Extensive docs and other sources
- A stable core

What takes developers years to learn, AI already knows.



Less effort

Gold-plating makes sense now

- Extra functionality
- Extra logging and reporting
- Extra edge cases handled
- CLI commands and admin settings
- Accessibility and usability polish



Pareto Principle

From 80/20 to 100/100

Effort is not a factor any more.

It makes sense to actually do 100% solutions now.

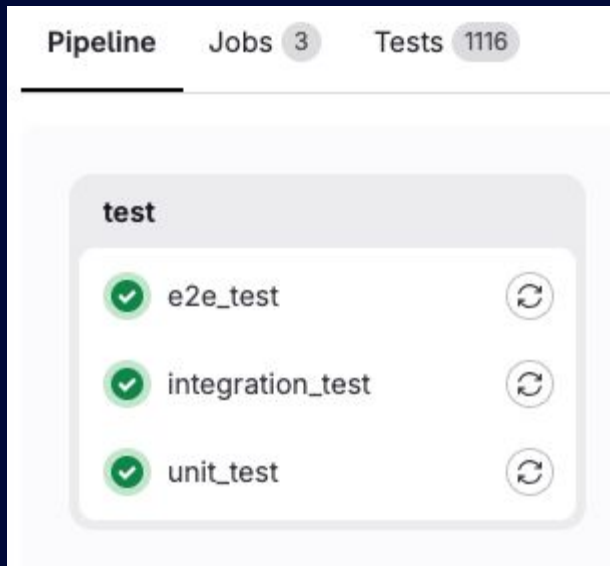
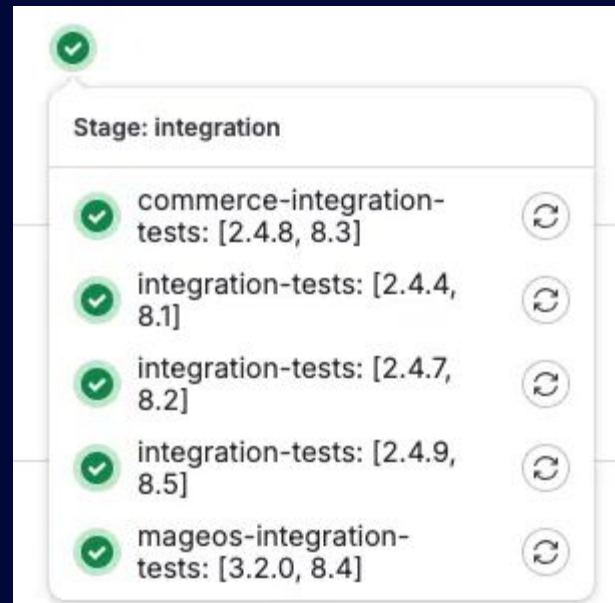


Lessons learned

Tests and CI got cheap

Full test coverage and CI are no longer a luxury.

- End-to-end tests
- Integration tests
- Unit tests



Lessons learned

Generated docs beat no docs

AI writes solid documentation.

Especially the parts nobody enjoys writing.

Custom skills / guidelines keep it in your style.

AI can write those skills, too.



hyvää

2. How to get there

What works for us



How to make AI create better code

Prerequisites

You need domain expertise.

You need technical expertise.

You need to be able to judge the results on all levels.

You need to accept ownership of the generated code.



Security

Sandbox it, then let it run

An OS-level sandbox limits what the agent can touch.

- Agent Safehouse for MacOS
- Other tools for other OS

Limit access to SSH keys

- 1Password @ Hyvä

Inside it, skip the confirmations and let it work.



Magento

Show AI the runtime

Bricklayer MCP exposes merged DI, plugins, layout and DB.

The AI looks things up instead of guessing.

Great for debugging too.



Lessons learned

Coding standards for free

AI follows coding standards by default.

With tools like phpcs in the loop, it fixes its own violations.



Tech Debt

Cleaning up

These tasks are cheap now, so they actually happen:

- De-duplicate code
- Restructuring
- Split 500-line methods and huge classes
- Transform to more modern syntax
- Make frontend code CSP compatible



Lessons learned

AI can use the Browser

In the browser, AI checks its own work and fixes what breaks.

The final manual test is still done by the developer.

"You're absolutely right!"

"Great catch! Let me fix that."

"Ah, I see the issue now."



TDD

Failing test first

Every bug starts as a failing test.

Every fix leaves a regression test behind.



Reviews

Keep the overview

Review all changes, at least roughly.

Don't review every line.

Review the architecture instead.

Don't hesitate to ask for big rewrites:
they cost minutes, not days.



Example

Security-critical code got safer

We cannot risk security issues.

AI reviews find gaps most humans won't find.

Still, security specialists matter more than ever before.

Humans decide what matters.



Hyvä code review skill

Seven reviewers, one per concern

- Accessibility
- Coding smells
- Copyright
- Grammar
- Performance
- Security
- Style



Lessons learned

Small change, big diff? Stop.

Interrupt.

Ask for the simple solution.

Let it run, but check in regularly.



hyvä

3. What changes for developers

A status report from inside the change





Discussion about
developers' roles at



Developers

The job is shifting

Less typing: scaffolding, boilerplate, routine tests.

More judgment: specifying, reviewing, spotting what's wrong.

More talking: with clients and the team.



Developers

Starting out today

The tasks juniors learned on are disappearing.

Experience still counts. Gaining it got harder.

No juniors today, not enough seniors in ten years.



Not embracing AI development yet?

It's not too late

Now is the best time to dive into AI assisted development.

First steps are easy.

An 18€ per month plan is enough for the start.

Use it as a tool. Everything else is optional.



10x development

Coding is 10x faster now?

Coding is.

Development is not.

2-5x is realistic if you care about quality.



More tasks in the same time?

Fight AI burnout

Don't start the fifth parallel task.

Take breaks.

Embrace token limits.



Prevent losing your skills

Continue learning

Do a manual task from time to time.

Let AI do what you already know
(or what you don't want to learn).

Do yourself what you don't know yet.



hyvä

4. Looking ahead

Personal Opinion



Back to the question

So AI makes code worse?

Only if you let it.



Why AI Code can be better than Human generated Code

Summary

AI became better a lot.

Go from 80% to 100% solutions.

AI can create and run automated tests easily.

AI does the things humans don't like.

AI reviews find more issues.

AI makes mistakes. Humans too.



ECOMMERCE MADE HAPPY 😊



Andreas von Studnitz

Senior Architect

@ andreas@hyva.io

in andreasv1