

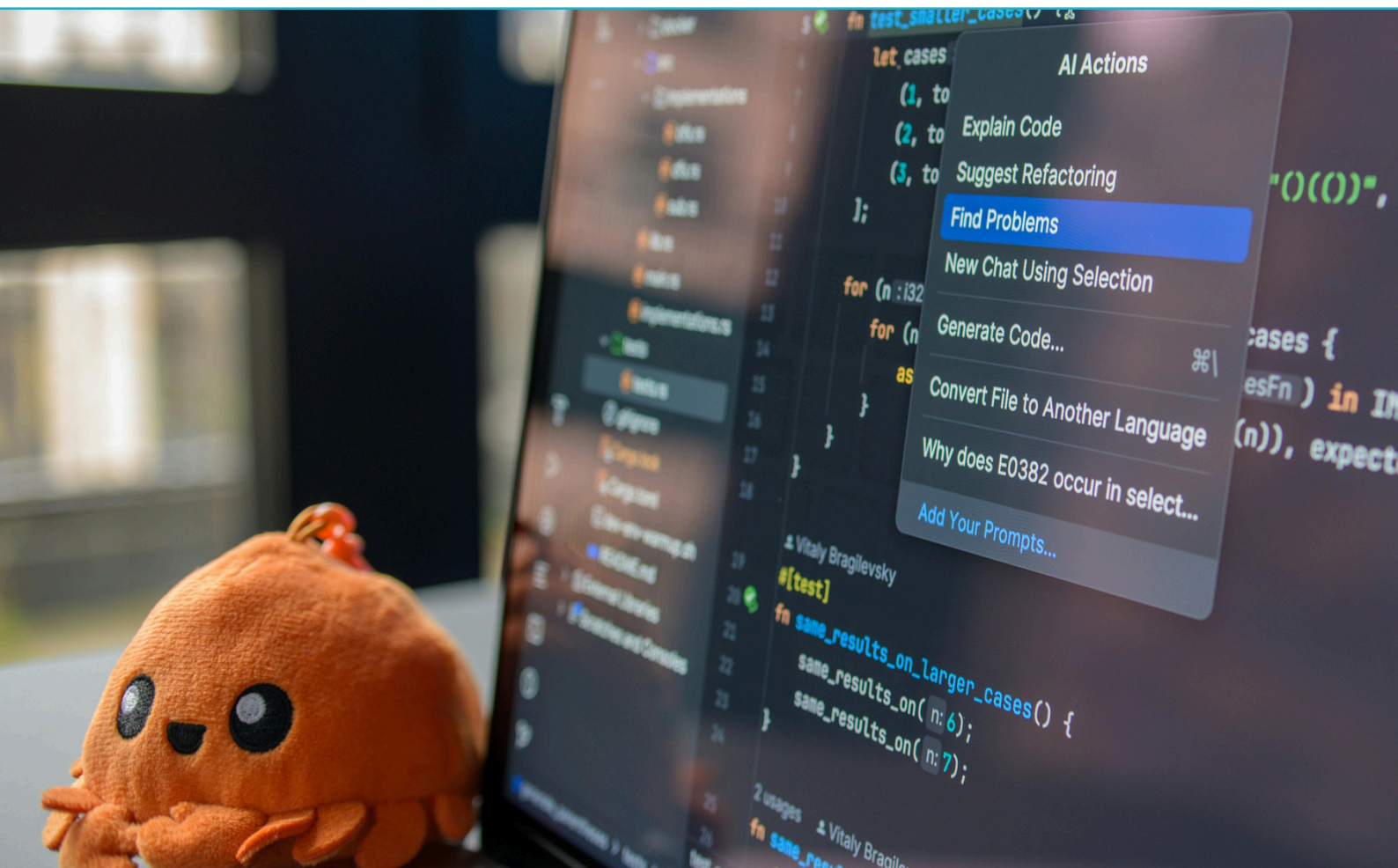
CASE STUDY

THE AI-ACCELERATED PIVOT

How Purity Health Switched Tech Stacks and Still Delivered



NASHVILLE  SOFTWARE SCHOOL



THE CHALLENGE

A Mid-Stream Pivot to Unfamiliar Territory

When Purity Health shifted their target to building an end-to-end clinical research platform, they faced a daunting technical crossroads. To build the solution they envisioned, the team needed to pivot their stack almost entirely, moving from C# and SQLPage to a more sophisticated Node/Express/Pug application. In the world of startups, there was no time for a slow ramp-up. The team needed to master the new stack—and ship—at the same time.

For their development team of two, this meant building an MVP using languages and frameworks they were less familiar with. Drew Goodman, a 2024 NSS graduate and junior developer at Purity Health, helped identify the need for specialized training in agentic AI. By enrolling their team in the Intro to Agentic AI for Developers class at NSS, the team equipped themselves to take advantage of generative AI in learning their new stack and Agentic AI tooling in developing their MVP.



THE LEADERSHIP PERSPECTIVE

Moving Faster, Safely

For CTO David Andrews and CPO Fletcher Watson, agentic AI was the key to maintaining momentum during the pivot. However, speed couldn't come at the cost of stability. They needed to implement strict guardrails to ensure the AI remained a tool and not a "runaway train."

"I see [AI] building itself deeply into the developer psyche... we encourage [our team] to write as much as possible with generative AI for the purpose of coding velocity... but your code review process has got to be very in depth and you have to have strong guardrails in place."



— David Andrews, CTO, Purity Health

"It is incredibly beneficial to have all of these checks and systems and philosophies for prompting shared amongst the team... seeing those two things [experience and curriculum] mirrored together was a really interesting observation."

— Fletcher Watson, CPO, Purity Health



THE DEVELOPER PERSPECTIVE

A Level Playing Field for All Experience Levels

The thing that made this pivot different from a traditional tech stack pivot was the simultaneous adoption of agentic AI, which was new to everyone. Whether a developer has five years of experience or five months, there was a learning curve. Developers Ripal Patel and Drew Goodman used agentic AI to bridge the gap between their prior experience and the new requirements of the Node stack. By moving beyond simple "chatting" and toward complex, agentic workflows, the team was able to master unfamiliar territory together and create code much faster than they could have alone.

The Intermediate View: Efficiency Through Planning

For Ripal, an experienced developer with seven years in the field, the transition was initially met with the common fear that AI might eventually replace her role. However, the agentic AI training helped her shift that mindset, moving from hesitation to openness to using AI as a tool for high-level logic and productivity. She now treats the AI as a partner in the planning phase, using an "ask mode" to vet her architectural designs and identify edge cases she might have otherwise missed. By letting the AI handle the initial heavy lifting of code generation, she can reallocate her time to rigorous code review, ensuring every feature meets production standards.



“Even though the tech stack was new, [AI] helped us to still continue in writing code and being more productive... and get the project moving along.”

— Ripal Patel,
Software Developer, Purity Health

The Junior View: Gaining System Insights

For Drew, the training provided a way to move beyond the typical hurdles of a junior developer, such as spending hours chasing a single missing semicolon. Instead of just being an "operator" who asks the AI to do the work, Drew learned that planning the design helped him keep a "leash" on the tool. This approach ensures the AI remains on track while helping Drew understand the logic behind the code it generates. By using AI as a learning assistant, he can focus more on how different parts of a project integrate, helping him become a more valuable developer faster.



“The more I talk to it like in pseudocode versus 'Hey, just do this for me'... not only is it more accurate, but it helps me learn and understand what I’m doing rather than just being an operator.”

— **Drew Goodman**
Software Developer, Purity Health

An Ever-Changing Toolkit

With the rapid growth and evolution of agentic AI, the team’s AI foundation has already been tested. Since their initial pivot, they have already dramatically overhauled their workflow to move from earlier agentic AI tools to Claude Code. This agility allows the team to stay at the cutting edge, ensuring their development process evolves as fast as the technology does.

KEY OUTCOMES

Velocity with Guardrails

By applying the concepts from the NSS Agentic AI for Developers class, Purity Health turned a complex pivot into a model for team-wide agentic AI use.

Accelerated Learning

By treating AI as a learning assistant rather than just a code vending machine, the team mastered the Node stack while building production features in parallel.

AI Guardrails That Scale Across The Team

Because different developers prompt AI differently and get different results, Purity Health created shared instruction files, "a README for the AI," that enforce consistent conventions regardless of who's writing the prompt.

A Shift from Writing to Reviewing Code

Because AI can make mistakes at a larger scale and faster pace than a human developer, the team's focus shifted. Ripal now spends the majority of her time reviewing AI-generated code rather than writing it from scratch, checking that it does exactly what was asked, no more and no less.

High-Fidelity Output

By maintaining a "human in the loop" for planning and code review, the team has achieved a workflow where agents operate deterministically, resulting in remarkably few regression bugs despite the speed of development.

Comprehensive, AI-Assisted Testing

Rather than relying on visual code reviews, the team requires a test for every new API call and every UI update before a PR can be merged. AI generates the tests, but developers review them with the same rigor as the code itself — including testing both what should work and what shouldn't.

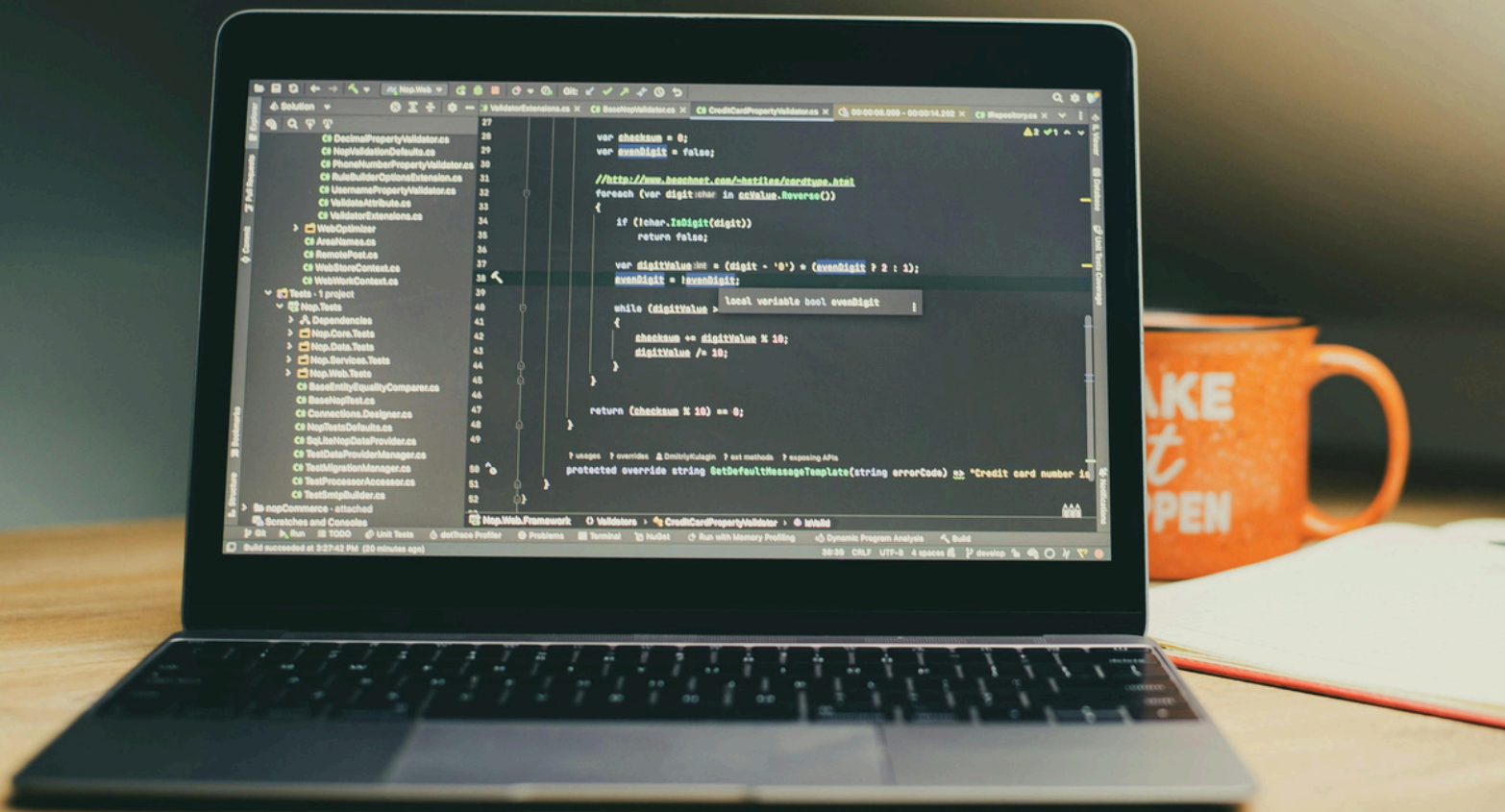
GitHub Guardrails in the CI/CD Pipeline

PRs automatically trigger the full test suite and flag unauthorized changes, like edits to package dependencies, before anything reaches production.

THE BOTTOM LINE

Training your developers in agentic AI doesn't just make them faster; it builds a more resilient team that's ready to pivot. By combining the "learning to learn" mentality NSS graduates learn during bootcamp with structured AI training, Purity Health was able to pivot their entire technical direction without missing a beat.

The team didn't stop after their initial training; they have expanded their proficiency by actively graduating from early AI prompts to deeper workflows like Claude Code. Building a team this way creates a technical foundation that can weather pivots and rapid industry shifts. And it will grow with you.



READY TO INCREASE YOUR TEAM'S VELOCITY?

Don't wait for the industry to move past you. Equip your developers with the skills to use AI safely, effectively, and at scale.



**ENROLL YOUR TEAM IN
INTRO TO AGENTIC AI
FOR DEVELOPERS**

NASHSS.COM/AGENTICAI