agoda

Agoda AI Developer Report 2025

How Software Engineers Work with AI Across Southeast Asia and India

Table of Content

Foreword

Idan Zalzberg
Chief Technology Officer

AI has quickly become part of how developers across Asia work and think. What began as a way to speed up tasks like writing, testing, or debugging code has grown into a broader shift in how software is built. Today, AI helps teams move faster, learn continuously, and solve problems in new ways.

This report captures that change. Based on survey data from developers in India, Thailand, Indonesia, Vietnam, Malaysia, the Philippines, and Singapore, it shows a region where AI is widely used but still maturing. Adoption is high, but reliability and structure are still catching up.

What defines Asia's developers is pragmatism. They view AI as a tool to assist, refine, and accelerate their work, not a replacement for skill or judgment. Gains in speed are matched by attention to quality. The result is a culture that moves fast but stays disciplined, pairing experimentation with careful review to keep standards high.

For organizations, the next step is to keep pace with this ground-up maturity. Success depends not only on adopting new tools but on building systems of trust and accountability around them.

This report looks at how that trust is developing and what it means for the next phase of AI maturity across developers in Southeast Asia and India.

Executive Summary

The findings in this report show how deeply AI is now embedded in the way developers across Southeast Asia and India build, collaborate, and learn, and how that rapid adoption is reshaping engineering practices across the region.

Developers are confident but cautious. Nearly **nine in ten** use AI each week, and about **three in four** expect long-term quality gains, yet only **two in five** trust AI to meet the quality of a mid-level developer today. That gap between enthusiasm and reliability defines the current stage: **mainstream, still mastering consistency**.

From this report, three interconnected findings emerge:

AI Is Mainstream but Not Mature

Use is universal, but structure and full-lifecycle integration trail adoption.

AI Is Evolving Through Accountability

Productivity depends on discipline; review and oversight are now embedded in how developers work.

AI Experience Is Uneven and Risks Creating Gaps

Differences in experience, company scale, and ecosystem readiness shape who scales fastest.

Across these findings, a consistent profile emerges: pragmatic optimism. Developers across southeast Asia and India are using AI with intent, to automate repetitive work, accelerate delivery, and improve quality through iteration. Where others chase automation at scale, the region emphasizes augmentation in depth, pairing speed with oversight, and efficiency with accountability.

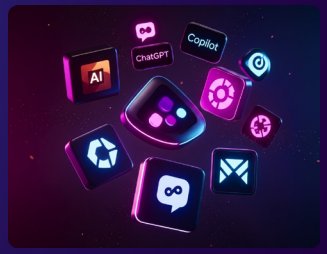
The next step is to transform what works in practice into common standards, enabling reliable methods to be repeated, shared, and scaled across the region.

Analytical Framework:

Five Themes Defining Developer Realities

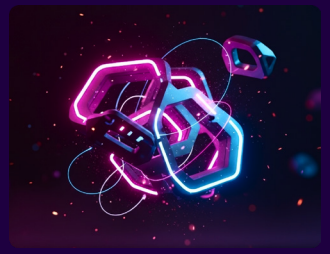
This report is based on Agoda's primary research with developers across Southeast Asia and India. The findings are organized into five themes that show how AI is changing daily development work:

Developer Mindset Towards AI



Tools & Stack Trends

AI in the Developer Workflow



Developer Productivity & Collaboration

Developer Talent & Growth

Together, these themes highlight a region where adoption is fast, use is practical, and maturity is still developing.

The next sections utilize this data to demonstrate the changes in how developers across Southeast Asia and India build, collaborate, and grow with AI.

01 Developer Mindset Towards AI

developers use AI every week

79% say inconsistent output is the biggest barrier

75% are confident in AI's long-term impact

43% believe AI can perform at mid-level engineer standard

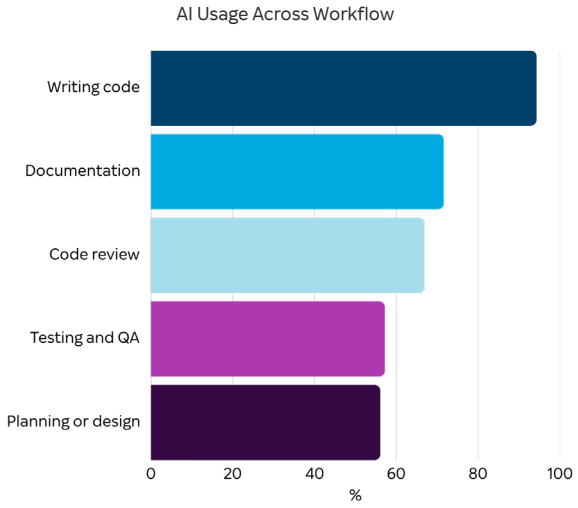
AI has become part of everyday development work. Most developers now use it weekly, with many keeping an assistant open at all times.

Productivity remains the anchor: **80%** cite speed and automation as their main motivation, while only **22%** use AI primarily to tackle unfamiliar problems.

Confidence, however, is split. Less than half believe AI can already perform at a mid-level engineer's standard; the rest remain unconvinced. Reliability stands out as the defining barrier, inconsistent or unreliable output limits adoption more than any other factor.

Despite these limits, optimism runs high. Most developers express confidence in AI's long-term impact, and **62%** believe it will expand career opportunities rather than reduce them. **71%** say AI already makes them better at their jobs, viewing it as a personal skill-builder rather than a threat.

The result is a pragmatic mindset that is optimistic, self-reliant, and anchored in personal experimentation, yet still cautious about reliability and governance.



Case Study:

SCB 10X

Scaling AI with Speed and Purpose

Company Context

SCB 10X, the innovation arm of Siam Commercial Bank, has made AI central to its engineering strategy. Its flagship initiative, Typhoon, Thailand's leading open-source large language model (LLM) built specifically for Thai language and cultural contexts, began as an internal productivity experiment and has since evolved into a national platform driving education, public-sector modernization, and enterprise AI innovation across the country.

Building with AI, Not Around It

SCB 10X adopted AI with one clear goal: accelerate innovation while maintaining quality. Led by Kunat Pipatanakul, Lead AI Scientist, the team integrated AI tools into every stage of the engineering lifecycle, from prototyping and testing to deployment. Their "Typhoon Application Week" proved what that integration could deliver: seven applications released in seven days, each built and deployed using Typhoon's in-house AI tooling.

Engineers now combine Typhoon with Copilot, ChatGPT, and Gemini for code generation, debugging, and quality control. Internal metrics show developers saving five to six hours per week through AI-assisted prototyping and code review. Typhoon's second generation adds larger context windows, multimodal inputs, and Thai-specific safety classifiers, all designed for real-world use rather than experimentation.

Culture and Adoption

AI adoption at SCB 10X is built on culture as much as capability. Leadership actively promotes experimentation, encouraging teams to test tools, share outcomes, and refine best practices. Every engineer is expected to explore AI methods and report what works, creating a continuous-learning loop across teams.

“Encourage a proactive culture,” says Pipatanakul — a phrase that has become shorthand for the company’s approach. The mindset is pragmatic: move fast, learn collectively, and institutionalize what works. Governance and experimentation coexist, ensuring that speed never outruns structure.

Challenges and Governance

The path hasn’t been friction-free. The high cost of AI tools remains the top challenge, with licensing and usage fees weighing heavily as adoption scales. Integration complexity ranks close behind, as teams coordinate multiple platforms and APIs across diverse systems. Governance and compliance also require constant attention, particularly when experimenting with models that handle sensitive or proprietary data.

To balance innovation with control, SCB 10X has introduced clear internal guardrails covering data use, prompt management, and model evaluation. These measures keep experimentation within safe bounds while preserving the agility that drives results.

Looking Ahead: From Velocity to Impact

Over the next 12–18 months, SCB 10X expects AI to move from support to core function, handling increasingly critical workloads. Pipatanakul anticipates that AI will soon be “able to do harder, mission-critical tasks.” Typhoon is already expanding beyond the bank, powering education initiatives with the Office of the Education Council and assisting public-service delivery through the Office of Public Sector Development Commission, reaching 9,700 students across 300+ schools and improving citizen-service workflows.

This trajectory aligns with Thailand’s national AI roadmap: home-grown models that strengthen local capacity and trust. Typhoon is more than a technical project, it’s infrastructure for Thailand’s digital future.

Why It Matters

SCB 10X shows how speed and responsibility can coexist. By embedding AI across engineering workflows and focusing on Thai language, context, and trust, the company is turning experimentation into national impact.

02 Tools & Stack Trends

87% use ChatGPT as their main AI tool

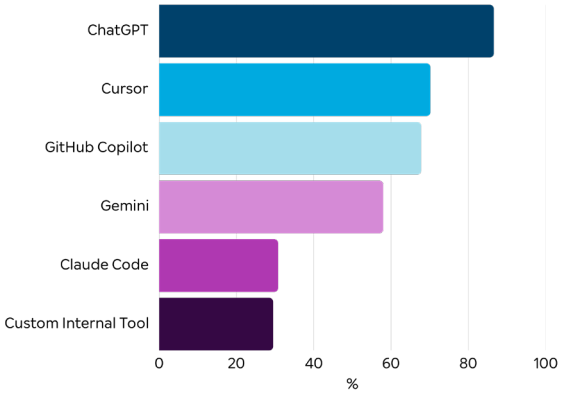
74% say IDEs are already AI-ready

60% of organisations have no formal AI policy

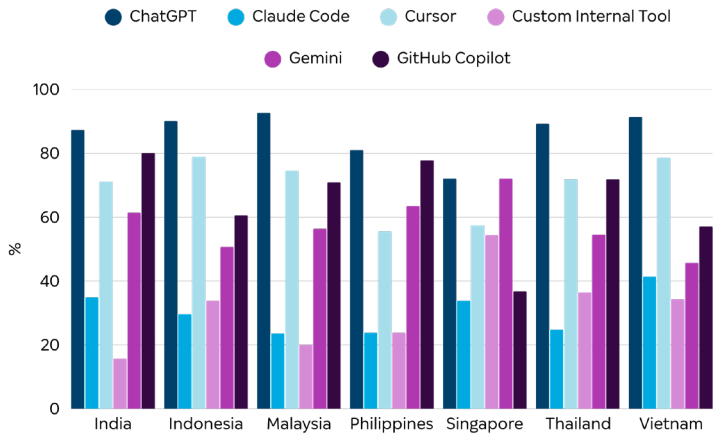
37% save 4-6 hours per week with AI

The AI toolchain across Southeast Asia and India is consolidating around a few dominant anchors. These AI tools are now part of most engineers' daily setup.

Top AI Tools Used



Top AI Tools Used - Country Breakdown



Even so, performance is inconsistent. Most respondents report that AI-generated code works less than half the time without major edits.

The productivity impact is real but modest. Most developers save between one and six hours per week. The biggest gains appear where tools already feel AI-ready: IDEs (**74%**) and deployment or infrastructure (**69%**), while readiness falls in testing (**34%**), code review (**20%**), and documentation (**16%**).

Governance remains an opportunity for companies. Many organizations lack a formal AI policy, creating an environment that operates at two speeds, with integration advancing faster than institutional maturity.

Case Study:

> Before we exposed AI to customers, we built confidence internally by understanding its limits, shaping guardrails, and training teams to use it responsibly.

Shaun Sit
Director of Engineering at Agoda

agoda

“Inside-Out” AI Strategy

Company Context

As a leading digital travel platform, Agoda operates at a global scale, with millions of searches, bookings, and data points each day. Integrating Generative AI into such a system demanded more than tool adoption; it required redesigning how the company learns, builds, and governs technology.

Building with AI, Not Around It

Agoda’s Inside-Out strategy began with a clear principle: build internal mastery before external exposure. Rather than rushing to launch consumer features, the company created a secure internal sandbox where engineers could experiment safely and securely.

At the centre is the GenAI Proxy, a unified access layer that manages governance, routing, and compliance.

Key Functions

- **Authenticates usage**
- **Enforces rate limits**
- **Monitors cost per request**
- **Provides a single compliance checkpoint**
- **Works across multiple LLM providers**

The impact is visible in scale: over 200 internal AI-powered tools now support engineering, design, and operations. Developers use them daily for code generation, debugging, documentation, and testing, improving overall development efficiency.

Example of Internal Utility

Internal Chat Assistant Platform (ChatGPT-style, in-house):

secure, in-house chat platform giving every employee access to GenAI assistants. Model-agnostic with a unified interface, it avoids per-seat licensing while letting Agoda define assistant behaviors centrally. Features include custom assistants, a prototyping playground, and a Chrome extension for contextual responses.

CodeBuddy:

Among Agoda's most impactful internal tools is CodeBuddy, an AI-powered assistant designed to automate and enhance the code review process. Its introduction has transformed development velocity, reducing the volume of human comments per merge request by more than 30%, a massive efficiency gain that directly accelerates time-to-market.

agoda

> This acceleration doesn't come at the cost of quality; CodeBuddy acts as a proactive partner, successfully catching and preventing real production bugs before they ever impact a user. And we know it's working because our team embraces it.

Royee Goldberg
VP of Engineering
at Agoda

Adoption has been positive, with a 90% approval rate among engineers, who see it not as a replacement but as a trusted collaborator that saves time while maintaining quality.

Challenges and Trade-Offs

Operating at scale brings unique hurdles:

Context engineering

retrieving accurate, relevant data from vast internal sources.

Scaling governance

ensuring each new AI tool meets compliance without slowing delivery.

Production readiness

turning rapid prototypes into resilient, audited services.

Agoda's answer has been architectural and cultural: automate checks through its Proxy layer while teaching developers to design with explainability in mind.

Looking Ahead: From Adoption to Autonomy

Agoda plans to expand its AI ecosystem into deeper product and customer-facing workflows.

Internally, the company continues to measure success through Adoption, Usage, and Time Saved, linking those metrics directly to engineering velocity and cost efficiency. As AI matures, Agoda expects its role to evolve from assistant to collaborator, an invisible layer powering every build, review, and release.

Why It Matters

Agoda's Inside-Out journey shows how large-scale platforms can adopt AI without losing control. By mastering governance before exposure, the company built trust from within, creating an ecosystem where innovation and integrity reinforce each other.

03 AI in the Developer Workflow

use AI for code generation

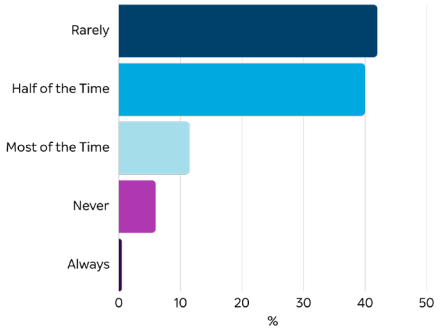
67% always review AI code before merging

72% say AI improves code quality

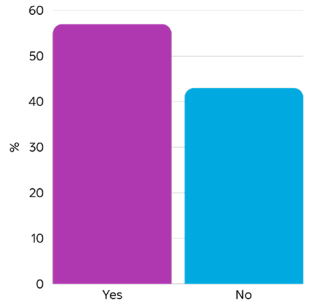
44% cite inconsistent results as main barrier

Developers now use AI across nearly every stage of their workflow, but most activity still centers on code generation.

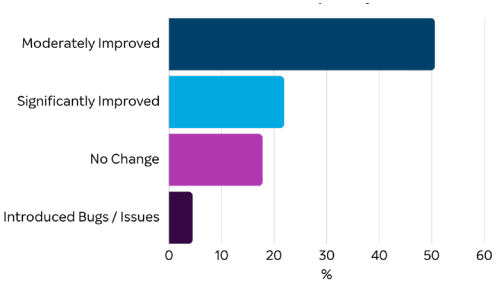
How often is AI-Generated Code Usable Without Significant Changes?



Should AI-Generated Code be Reviewed Differently to Human-Generated Code?



How has AI affected your ability to deliver quality code?



Most developers see AI as an accelerator, not a replacement. Most say it improves code quality. However, reliability remains a challenge; inconsistent results remain the primary barrier to wider adoption.

Teams are adapting by building review habits into their workflow. Most always review AI-produced code before merging, and nearly **70%** routinely rework it for correctness. Oversight has become a habit rather than a policy, a bottom-up audit culture that values verification as much as velocity. Still, only a minority of teams have codified these behaviors; formal review frameworks remain rare.

AI use has become routine, but it is not yet structured. Developers across Southeast Asia and India have built an instinct for verification; the next step is to turn that instinct into shared practice.

Case Study:

Integrating AI End-to-End: Path to Productivity and Trust

Company Context

M_Service, the operator of MoMo, is Vietnam's leading super app with more than 31 million users. As its platform expanded across payments, financial services, and consumer utilities, the company's engineering organization had to scale quickly. That growth made it an ideal environment to explore how AI can redefine developer productivity at scale.

Building with AI, Not Around It

At M_Service, the team led by Nam Dinh, Head of the AI for Productivity Development project, has embedded AI into daily engineering practice to make teams faster, smarter, and more focused on high-value work.

The company deploys a suite of commercial and proprietary AI tools, including GitHub Copilot, ChatGPT, Claude, Gemini, Cursor, Windsurf, and its internally built Model Context Protocol (MCP) connectors, to securely link private and partner systems. Real results are already visible:

Figma2Code

converts designs to React Native components, meeting 95 % of internal standards, cutting review time from 2–3 hours to 30 minutes.

Code Review Assistant

detects code smells and security gaps across pull requests, reducing review cycles by 40 %.

Project Intelligence

connects with Jira to auto-generate reports, saving managers 3–4 hours weekly.

Automated QC and Test Case Generation

raises test coverage from 70% to 90% and halves authoring time.

M_Service's generate–review–refine rhythm captures how disciplined workflows turn experimentation into reliability.

Culture, Not Just Code

Technology alone doesn't guarantee success. Nam Dinh credits culture:

"We operate with the slogan 'AI First,' which fundamentally shapes our organizational DNA."

Leadership encourages continuous experimentation, while diverse engineering backgrounds, mobile, backend, systems, help identify where each AI tool adds the most value. That ethos turns curiosity into structure, mirroring the regional pattern of transforming early adoption into repeatable, accountable practice.

The Hard Parts

Progress hasn't been seamless. Key challenges include:

Rapid tool evolution

optimization often outpaced by the next release.

Tool stability

many remain in beta, with inconsistent uptime.

Integration complexity

orchestrating multiple AI tools without disruption.

Scale

rolling out change across large, distributed teams demands coordination and training.

Even with strong leadership, maintaining alignment across evolving platforms is non-trivial. Yet the company's AI governance focus ensures experimentation doesn't come at the expense of quality.

Looking Ahead: From Assistance to Autonomy

M_Service aims to weave AI through the entire software-development lifecycle, from product planning and design to testing, deployment, and operations, with a target of 40–50 % automation of repetitive tasks within 12 months and a full ecosystem within 18 months.

Planned advances include:

Product phase

Analyzing user behavior to propose features and automate research.

Development phase

many remain in beta, with inconsistent uptime.

QC phase

orchestrating multiple AI tools without disruption.

Operations phase

rolling out change across large, distributed teams demands coordination and training.

This roadmap blends innovation with accountability, precisely the maturity shift Agoda identifies as defining the region's next phase.

Why It Matters

M_Service shows that sustainable AI adoption depends as much on structure and culture as on tools. By pairing an "AI First" mindset with disciplined governance, the company turns productivity gains into institutional capability.

04 Developer Productivity & Collaboration

say AI makes them faster

73% report better code quality with AI

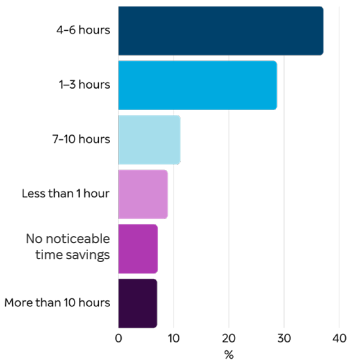
27% say teamwork has improved

39% report significant time savings

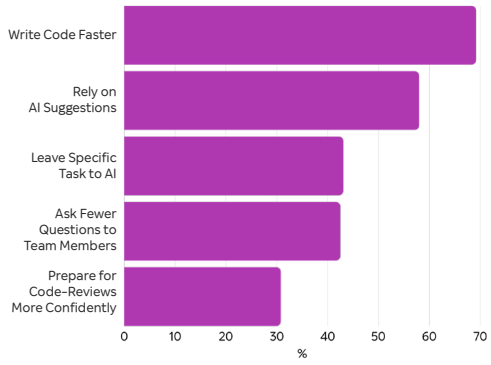
Speed is the universal dividend of AI. Nearly all developers say AI makes them faster, with many citing significant time-savings and others describing moderate gains. Quality also improves for most, though teams describe the lift as incremental rather than transformative. Friction is limited: only **36%** report any confusion or overlap in responsibilities, yet the experience gap is visible; **55%** of junior developers feel that friction compared to **33%** of seniors. Across markets, differences are minimal: in every country, more than nine in ten developers report speed gains.

But collaboration is still catching up. Teamwork practices have improved for only a minority of developers. Peer review is evolving, **57%** say their code-review process has changed, but leadership involvement lags behind, with fewer than half of managers (**46%**) actively encouraging AI experimentation. The productivity dividend remains personal, not institutional. AI is driving efficiency faster than it is redefining collaboration.

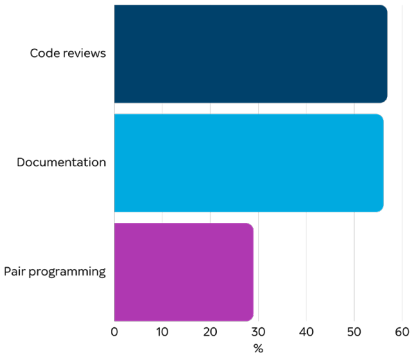
Time Savings Per Week Using AI in the Workflow



Shifts in Team Productivity



Largest Impacts on Team Practices Due to AI



Case Study:

OMISE

Engineering Trust Through AI Precision

Company Context

Omise operates trusted payments infrastructure across Asia. In a domain where reliability is non-negotiable, the team's approach to AI is practical and quality-centred. Sylvain Dormieu, Director of Engineering, leads with a simple premise: use AI to reinforce standards, not relax them.

From Speed to Certainty

Omise has introduced AI where it directly strengthens delivery discipline. Engineers use Cursor, GitHub Copilot, and other tools such as Claude Sonnet, GPT, and Gemini for scaffolding, test snippets, and refactoring.

Early pilots focused on PR requests within the development flow to ensure tasks were completed with proper review and that sensitive data remained secure. After trialling multiple systems, the company standardised on a licensed setup to track usage, balancing exploration with governance.

Culture and Governance

The culture is intentionally hands-on, with engineers encouraged to try tools while leaders observe outcomes and usage patterns. There's no single ROI metric yet; value is judged by usage signals and qualitative feedback from engineers.

> The conversation about AI isn't separate from the conversation about quality.

Sylvain Dormieu
Director of
Engineering Omise.

Skills emphasis has shifted

Code reviews, unit tests, and QA now carry more weight as AI enters the workflow. This evolution has been about upgrading practice rather than reducing people, no headcount cuts accompanied the shift.

Software engineering, as Dormieu frames it, is about implementing workflows that reflect how engineers think. Readability and maintainability remain paramount. With AI tools, developers stay accountable for the code they produce, they're simply faster at producing it.

Internally, visibility and accountability drive adoption. Engineers share prompt examples and review insights in open channels, turning individual experimentation into collective progress. Small "AI sessions" are run within teams to demonstrate new workflows and compare outputs, helping to build trust in the process while normalising peer validation.

Collaboration and Craft

Dormieu stresses mindful use. Copy-pasting AI output without context "shows little thinking" and risks drifting out of context; the expectation is review-by-default.

Teams trialed multiple systems before settling on their current stack, and knowledge sharing focuses on what reliably improves throughput without eroding standards.

That balance between freedom to explore and discipline to review defines Omise's engineering rhythm. It keeps curiosity alive while anchoring experimentation in traceable, explainable code.

Challenges and Trade-Offs

The limits are clear: AI can go wrong, and it can generate excess code, increasing the review load. In the world of payments, where accuracy and compliance are non-negotiable,, trust is earned through checks, not assumed.

Dormieu highlights the need for tools that:

- **Validate implementation design before generation, ensuring structural soundness.**
- **Detect wrong or forgotten patterns early, reducing breaking changes.**
- **Balance productivity with review discipline, so speed never overrides safety.**

Looking Ahead: From Assistance to Assurance

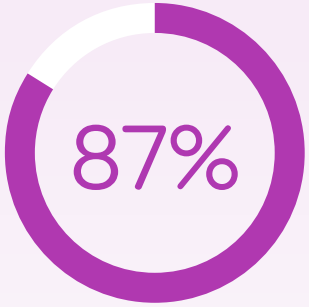
Looking ahead, the next 12–18 months will bring broader AI integration beyond code itself. Internal tools are expected to multiply, extending into adjacent engineering processes. As Dormieu notes, in a world where prototyping is effortless, any manager can create a tool to support their work in ways that once took months. This shift could unlock new productivity through greater creativity and experimentation across teams.

Omise is also exploring ways to quantify reliability gains, tracking time saved in reviews and the consistency of outputs, to ensure AI progress translates into measurable software quality.

The company expects explainability to become a differentiator, investing in methods that make AI-assisted logic transparent for future audits.

The trajectory is consistent: expand usage where review and reliability are transparent, keep governance light but visible, and favour improvements that tighten quality loops rather than chase autonomy for its own sake.

05 Developer Talent & Growth



changed learning priorities due to AI

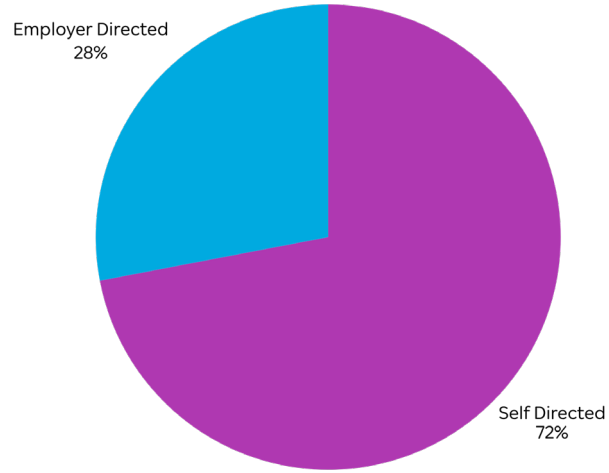
AI is reshaping how developers learn and grow, with most seeing expanded opportunities ahead. Upskilling is overwhelmingly self-directed: most self-train through tutorials, side projects, or experimentation, while only a minority receive employer-led instruction.

72% are self-taught in AI skills

58% believe AI proficiency should be a hiring requirement

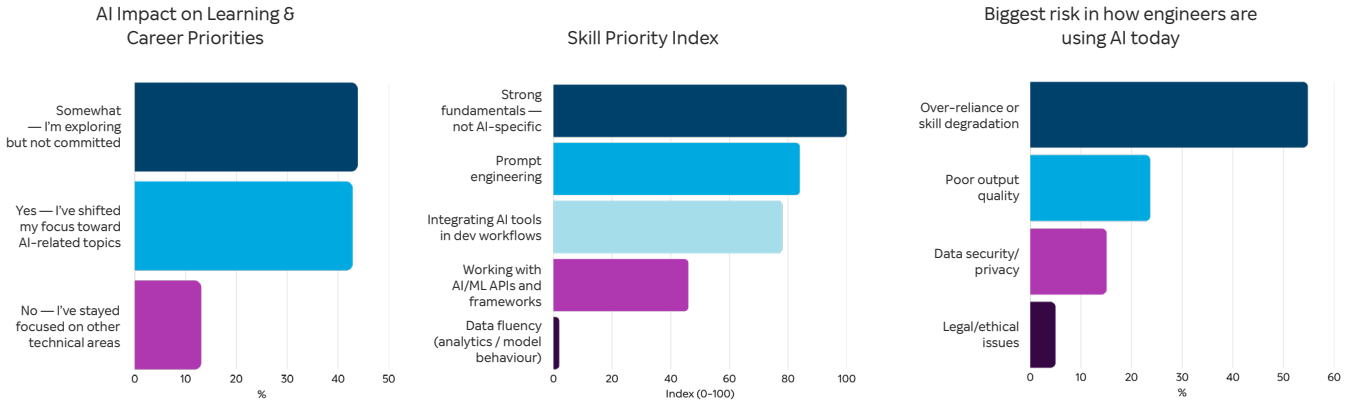
28% receive employer-led AI training

How Developers Are Learning AI Skills



58%

of respondents say AI proficiency should be a hiring requirement



Peer learning fills the gap; **52%** rely on online communities or open-source projects as their main development channel. Access to structured training varies sharply by market: in Singapore, developers are almost twice as likely as those in Vietnam to have formal AI learning programs available. The result is a workforce learning faster than organizations can teach, self-propelled, uneven, and increasingly independent.

Alongside that ambition comes pressure. **44%** of developers worry about falling behind if they don't keep up with AI's pace of change, a figure that climbs to **51%** in the Philippines and **44%** in Singapore. Expectations are also rising: most now believe AI proficiency should be a baseline hiring requirement. The 25-point confidence gap between senior and junior engineers highlights how access to mentorship and training can accelerate skill development and close experience gaps.

Together, these five themes show a region where adoption is fast but uneven. Developers are confident and moving quickly, but structure and formal support are still catching up.

The next section examines what this means in practice: how developers are using AI on a day-to-day basis, where discipline drives results, and where maturity still needs to grow.

Case Study:

> Our goal is to make every interaction easier and more trustworthy, to help people transact with confidence.

Rajath Ramesh
Group Director,
Product & Platform
Engineering at Carousell

carousell

Human-Centered AI for Smarter Marketplaces

Company Context

Carousell, one of Southeast Asia's largest classifieds and recommerce platforms, connects tens of millions of buyers and sellers across seven markets. With millions of listings and messages generated daily, its engineering teams face a constant challenge: keeping marketplace interactions simple, safe, and fast.

Building Smarter Marketplaces with AI

Carousell's engineering organization has embedded AI across multiple layers of its platform:

Image intelligence

automates product categorization and quality detection, flagging spam or mislabeled listings instantly.

Conversational AI

improves buyer–seller chats by identifying potential scams or sensitive content early.

Recommendation engines

refine relevance by learning from user behavior, improving discovery and conversion.

4-5 Hrs

saved per engineer weekly

> The biggest gain isn't speed alone it's time to think. AI gives us back hours we can use for design, problem-solving, and mentoring.

Rajath Ramesh
Group Director,
Product & Platform
Engineering at Carousell

Culture of Collaboration

Carousell's approach to AI reflects its product philosophy: community-driven, transparent, and iterative. Developers are encouraged to experiment and share findings through open Slack channels and weekly "show-and-learn" sessions.

Ramesh describes this as "a culture of shared discovery." Teams contribute prompts, code snippets, and test results into shared repositories, turning individual learnings into collective progress. Leadership enables rather than dictates, instead of rigid policies, Carousell provides clear guardrails around safety, privacy, and fairness, allowing teams to innovate responsibly within trusted boundaries.

This balance of autonomy and accountability has made experimentation both fast and disciplined, creativity supported by structure.

Challenges and Trade-Offs

Despite strong adoption, Carousell faces familiar hurdles:

Bias and safety

ensuring generative outputs meet brand and regulatory standards across diverse languages.

Scalability

maintaining model accuracy and performance as marketplace data grows.

Integration friction

blending new AI systems with existing infrastructure and analytics pipelines.

Ramesh refers to the company's approach as "guardrailed velocity." Lightweight review mechanisms, bias testing, and internal audit loops keep innovation moving without compromising user trust. The philosophy is clear: move fast, but never at the expense of reliability.

Looking Ahead: From Assistance to Autonomy

The next phase of Carousell's AI roadmap focuses on contextual intelligence, systems that understand marketplace nuance, from product types and pricing to local slang and intent. Internal APIs now connect listing text, image data, and user behavior to deliver more adaptive, self-improving recommendations.

Ramesh is also leading efforts to formalize AI-readiness training across teams. The goal is to ensure consistent capability so AI becomes a dependable layer of productivity, not a variable one. "We're building a bridge between human creativity and machine assistance," he says. "Developers should focus on what's new, AI can take care of what's known."

Why It Matters

Carousell shows how trust and technology can scale together. By embedding AI across both product and engineering while keeping people firmly in the loop, the company proves that innovation doesn't mean removing humans; it means amplifying them.

AI Is Mainstream but Not Mature

Across Southeast Asia and India engineering communities, AI has moved from experiment to expectation. Adoption is almost universal, and most developers use AI assistants as part of their regular workflow. The barrier is no longer access but alignment: how to turn frequent use into consistent performance. Confidence is high, but reliability divides opinion. Most developers say AI output still needs revision before it meets production standards.

The productivity gains are steady but measured. Most engineers save a few hours each week, though much of that time is spent reviewing or testing AI output. Teams that include review early in their process report fewer bugs and higher confidence in their results.

What emerges is a portrait of pragmatic maturity. Southeast Asia and India developers are not chasing novelty; they are engineering reliability. Progress is measured in consistency rather than hype. AI has become mainstream by habit but not yet by mastery. The region's developers are not holding AI back; they are holding it accountable and that rigor is the foundation of real maturity.

AI is Evolving Through Accountability

The first stage of AI maturity was adoption; the second is accountability, a phase where developers are evolving their practices through discipline and review. Developers now trust AI enough to accelerate work but still validate every step. Usage remains strongest in code generation and early development but drops in testing and deployment. The same tools that boost speed now demand consistent validation.

Most of that validation happens from the ground up. About two-thirds of developers always review AI-generated code before merging, and most make additional edits for quality. Few teams have written guidelines, and only a minority of managers actively promote AI experimentation, so governance for now is cultural rather than structural.

This culture of accountability is spreading laterally. Developers share prompts, critique each other's methods, and normalize review-by-default. Responsibility for quality is distributed: the reviewer owns the merge, and oversight is collective rather than top-down. What once appeared to be micromanagement now serves as shared risk management.

Developers in Southeast Asia and India are entering a phase of confident discipline, evolving their workflows through accountability. They rely on AI to speed up delivery but rely on human oversight to keep quality high. The next step is to turn this evolving culture of review into measurable systems that scale across teams and organizations.

Oversight isn't optional, review discipline is part of getting value. The region's developers are not holding AI back; they're holding it accountable. Their rigour is the foundation of real maturity.

AI Experience Is Uneven and Risks Creating Gaps

Now that adoption is almost universal, the challenge in Southeast Asia and India is no longer about who uses AI, but how effectively it's used. The tools are the same, yet outcomes differ widely. The gap lies in how teams learn, govern, and scale their use of AI.

Most developers are self-taught. 71% learn through tutorials, side projects, or experimentation, while fewer than one in three receive structured training from their employers. Senior engineers often set the example, but where mentorship is weak, adoption becomes inconsistent. Company size and market conditions also shape progress: start-ups move quickly with fewer restrictions, while larger enterprises advance more slowly under formal policies. India and Malaysia lead through community learning, Vietnam and Thailand sit in the middle, and Indonesia and the Philippines trail due to limited access and resources.

Unevenness, however, is not weakness. It maps opportunity. Later adopters can build on proven practices instead of repeating early challenges, while regional and global teams can share best practices across markets. The next phase of maturity will depend less on access and more on enablement, helping teams turn individual progress into collective capability.

Targeted enablement beats one-size-fits-all. Southeast Asia and India's uneven AI landscape is not a gap to close but a gradient to climb — one market, one team, and one developer at a time.

Southeast Asia and India's uneven progress isn't unique, it reflects a broader global pattern in how developers adapt to AI. What sets the region apart is its approach. While many global peers focus on structure and compliance, developers in Southeast Asia and India emphasize speed, experimentation, and learning through use. That balance gives the region a practical advantage as AI adoption continues to mature worldwide.

Southeast Asia and India in Global Context

Southeast Asia and India's developer markets are no longer peripheral to the global AI story; they are shaping its next chapter. Compared with leading global studies, the region moves faster, adapts earlier, and often sets its own rules.

Global optimism toward AI is widespread, but the version in Southeast Asia and India is more grounded. While most developers globally view AI positively, 79% in Southeast Asia and India say inconsistent results remain their biggest barrier to adoption. Developers in the region value speed but demand reliability before trust.

Tool preferences reflect this pragmatism. JetBrains 2024 reports 77% of developers globally use ChatGPT; Agoda's dataset shows 87% in Southeast Asia and India. GitHub Copilot follows similar patterns, 46% globally versus 68% regionally. This reflects different priorities: global enterprises emphasize integrated IDE workflows, while Southeast Asia and India prioritize accessible, flexible tools that support diverse languages and rapid experimentation.

The same contrast appears in skills. McKinsey's 2025 report states that 84% of global employees receive structured AI training, whereas across emerging Southeast Asia and India, only 28% do. Yet 87% self-train through tutorials and side projects. This independence can appear risky from the outside, but it often drives agility and faster skill development.

Southeast Asia and India's progress combine scale with speed; mainstream adoption is achieved before full maturity. The next step is not to copy global governance models but to create frameworks that maintain that momentum while improving trust, quality, and accountability.

Southeast Asia and India aren't catching up to global AI maturity — it's redefining how maturity happens: faster, flatter, and fuelled by proof over policy.

The comparison shows that Southeast Asia and India's strength lies in speed and adaptability. The next step is to pair that momentum with stronger foundations in trust, quality, and governance. The following section outlines how industry, enterprises, and policymakers can make that shift.

Implications for Southeast Asia and India Developer Ecosystem

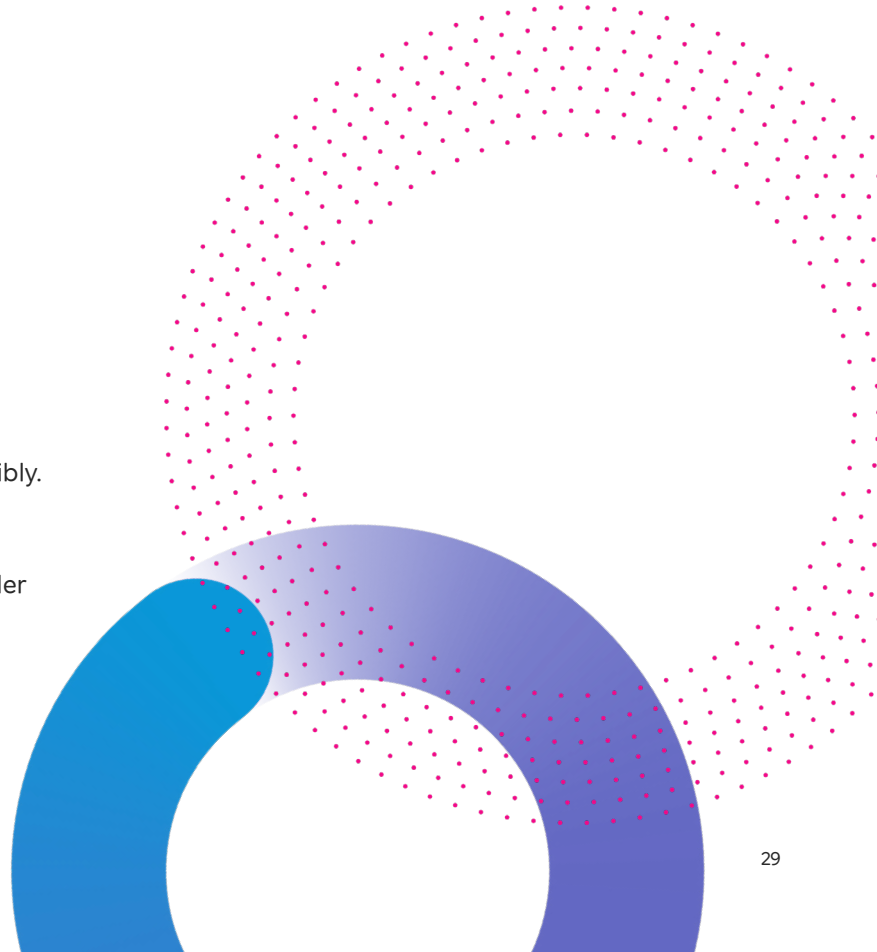
The developer ecosystem in Southeast Asia and India is at a turning point. Artificial intelligence is no longer about access, it's about maturity: how effectively organizations turn everyday use into dependable results. The question now is simple: how can we turn adoption into lasting impact?

1. Industry-Level Implications: Building Collective Trust Infrastructure

AI adoption in Southeast Asia and India has grown faster than the frameworks needed to manage it responsibly. At the industry level, the next step is to create shared trust infrastructure, open benchmarks, certification pathways, and interoperability standards that maintain quality without slowing innovation. Governments and professional associations can support this effort by funding regional sandboxes, developing cross-border audit frameworks, and reducing market overlap.

For a region defined by diversity and uneven maturity, coordination will matter as much as capability.

If Southeast Asia and India develop its own governance tools, it can define what "responsible AI" means for the global engineering community.



2. Company-Level Implications: Turning Oversight into a Productivity Function

Developers have already normalized review culture; enterprises must now meet them there.

Around 60% of organizations report no formal AI policy, meaning most productivity gains remain personal rather than shared. Standardizing simple review steps, peer checks, ethical-use guidelines, and clear escalation points can turn informal diligence into reliable governance.

Good governance doesn't slow teams down; when it's clear and consistent, it becomes a driver of productivity rather than a barrier. Companies that build accountability into everyday work will scale AI use more effectively.

3. CTO & Engineering Leadership Implications: Designing for Accountability

For engineering leaders, the primary challenge is striking a balance, allowing teams the freedom to experiment while maintaining quality, integrity, and privacy.

Most AI use still happens at the individual level, with only about a quarter of teams working under clear governance. Leaders can close this gap by building simple guardrails that define how AI-generated outputs move into production, encouraging senior engineers to guide juniors through code reviews and prompt design, helping close the confidence gap in using AI effectively and also tracking both speed and accuracy, linking productivity gains to quality improvements.

4. Policymakers: Enable Responsible Growth

Policymakers can strengthen regional collaboration by aligning data-sharing rules, funding open testing frameworks, and supporting shared ethical-AI research. Coordinated policies will help Southeast Asia and India developer community scale AI responsibly across borders.

5. Strategic Opportunity

The overarching implication is that unevenness is opportunity. Self-taught developers, peer mentorship, and start-up agility have created a dynamic foundation. By formalizing what developers already do well, the region can move toward maturity more quickly. The region's engineers have proven that AI can accelerate output. The next step is to scale these gains responsibly and inclusively.

The next differentiator will not be who adopts AI first, but who builds clear and consistent governance around it.

Momentum built on trust becomes the advantage. Southeast Asia and India's strength have never been conformity; it's the confidence to build systems as fast as ideas.

Methodology

This research was conducted via an online survey between August to September, 2025 gathering responses from over 600 developers across seven key markets in Southeast Asia and India: Indonesia, Malaysia, Singapore, Thailand, the Philippines, Vietnam, and India. Participants spanned a mix of experience levels, company sizes, and industry sectors, providing a representative view of how AI is being adopted, integrated, and experienced across the region's developer ecosystem.

As a leading digital travel platform exploring the role of AI in developer workflows, Agoda conducted this study to understand not only adoption and productivity trends, but also how developers are learning, collaborating, and building long-term AI capability. The survey examined five key themes: developer mindset, tools and stack trends, AI in the workflow, productivity and collaboration, and talent and growth, with findings supplemented by case studies from regional companies to illustrate real-world applications and trends.

Primary Data Sources

Agoda AI Developer Survey 2025

The survey was commissioned by Agoda and conducted in partnership with Macramé Consulting to understand how engineers across Southeast Asia and India work with AI

External Research

1. **Stack Overflow** – 2024 Developer Survey
2. **JetBrains** – The State of the Developer Ecosystem 2024.
3. **McKinsey & Company** – Superagency in the Workplace 2025
4. **PwC** – Asia Pacific Hopes and Fears Survey 2024

agoda

Join Engineering & Technology at Agoda

Build the future of travel with us.

Scan to view open roles