AI-NATIVE SDLC · PART 1 OF 3

Four companies. Four AI-native SDLCs.

What each says about building software once agents write most of the code. Where they agree, where they really disagree, and how much is measured.

- Anthropic
- Atlassian
- Google Cloud
- Shopify

Jinto Jose · personal research

Swipe →

FOUR ALTITUDES

Not four answers to one question. Four different questions.

ANTHROPIC · PLAYBOOK

What artifact does each stage commit?

6 stages, each ends in a file the next stage reads.

ATLASSIAN · PLAYBOOK

What must the platform provide?

5 workflow shifts on a context graph and measurement.

GOOGLE CLOUD · MARKETING PAGE

What can AI do in each phase?

A capability map, plus risks and best practices.

SHOPIFY · CASE STUDY

What happens when a whole company goes AI-first?

Five years of what worked, what broke, what it cost.

SETTLED GROUND

Five things all four agree on

1. **Code generation is no longer the constraint.**
   The bottleneck moved to planning, review and maintenance.

2. **Human review becomes the binding constraint.**
   Reached independently by three of the four.

3. **Context quality decides output quality.**
   Not model choice.

4. **Humans stay accountable for judgement.**
   Agents execute and enforce. People decide and own.

5. **It's an operating-model change.**
   Not a tool purchase.

STAGE BY STAGE

Same lifecycle, four treatments

Anthropic · Atlassian · Google · Shopify

	Anthropic	Atlassian	Google	Shopify
Plan	intent.md, in the asker's words	AI drafts specs from code + customers	NLP turns feedback into requirements	Non-engineers build own tools
Design	spec.md, constrained by org skills	Designers ship UI code	Pattern + compliance advice	Prototypes, not mockups
Build	plan.md approved before code	Context compounds in a graph	Boilerplate absorbed	Multi-tool by choice
Test	Continuous evals, 20–50 tasks	*No Test stage, on purpose*	Generated edge cases	Reversion rates unchanged
Deploy	Hooks as gates; agent can't self-approve	AI reviews first	Cloud Deploy + monitoring	Human review still mandatory
Maintain	Breaches write the next intent.md	Agents triage routine work	ML anomaly detection	Comprehension debt is the risk

WHERE THEY PART WAYS

Three real disagreements

1. Where does the source of truth live?



Expensive to reverse.

2. One standard tool, or many?



Governance appetite decides, not engineers.

3. How hard are the limits?



Regulated firms land left. Know the friction cost.

MEASURED OR ASSERTED?

Not one publishes a methodology.

Atlassian

STRONGEST NUMBERS

+44% accuracy · −48% tokens · Cycle time 11.5 → 6.7 days · 51% vulns auto-resolved

Self-reported internal benchmarks.

Shopify

MOST CREDIBLE CONTEXT

~20% productivity · Reversion rates unchanged

Its engineering head later said the 20% was his own rough number.

Anthropic

BEST MEASUREMENT DESIGN

No outcome figures · An indicator per play

Tells you what to measure, not what they measured.

Google Cloud

WEAKEST EVIDENCE

No figures · No customers · No research

GOVERNANCE

What a regulated organisation gets from each

DIMENSION	ANTHROPIC	ATLASSIAN	GOOGLE	SHOPIFY
Audit trail	●●●	●●●	●	●
Enforceable gates	●●●	●●	●	●
Separation of duties	●●●	●●●	●	●●
Data residency	●●●	●●	●●	●
Measurement	●●	●●●	●	●●
Change-management fit	●●●	●●	●	●

My reading of the published material, not a vendor assessment. ●●● = explicit, implementable guidance.

NOT A WINNER

Take one thing from each

ANTHROPIC

The artifact chain. Skills advise, hooks block.

Every stage commits a file the next stage reads. The commit history is the audit trail.

ATLASSIAN

Context as a platform. Measure before you scale.

So you can tell real gains from hype.

GOOGLE CLOUD

Generated code gets no review exemption.

Peer review and security scanning stay mandatory.

SHOPIFY

Nudge, don't mandate. Protect comprehension.

Make usage visible. Match the tool to the user.

• YOUR MOVE

Nobody disagrees on the destination. Three decisions are yours.

1 **Repo or platform graph as your source of truth?**

2 **One agent surface, or many tools on one layer?**

3 **Hard gates, or alerts that keep experiments alive?**

Full research, sources and a 16-point self-assessment:

github.com/jintomjose/ai-native-sdlc-playbooks