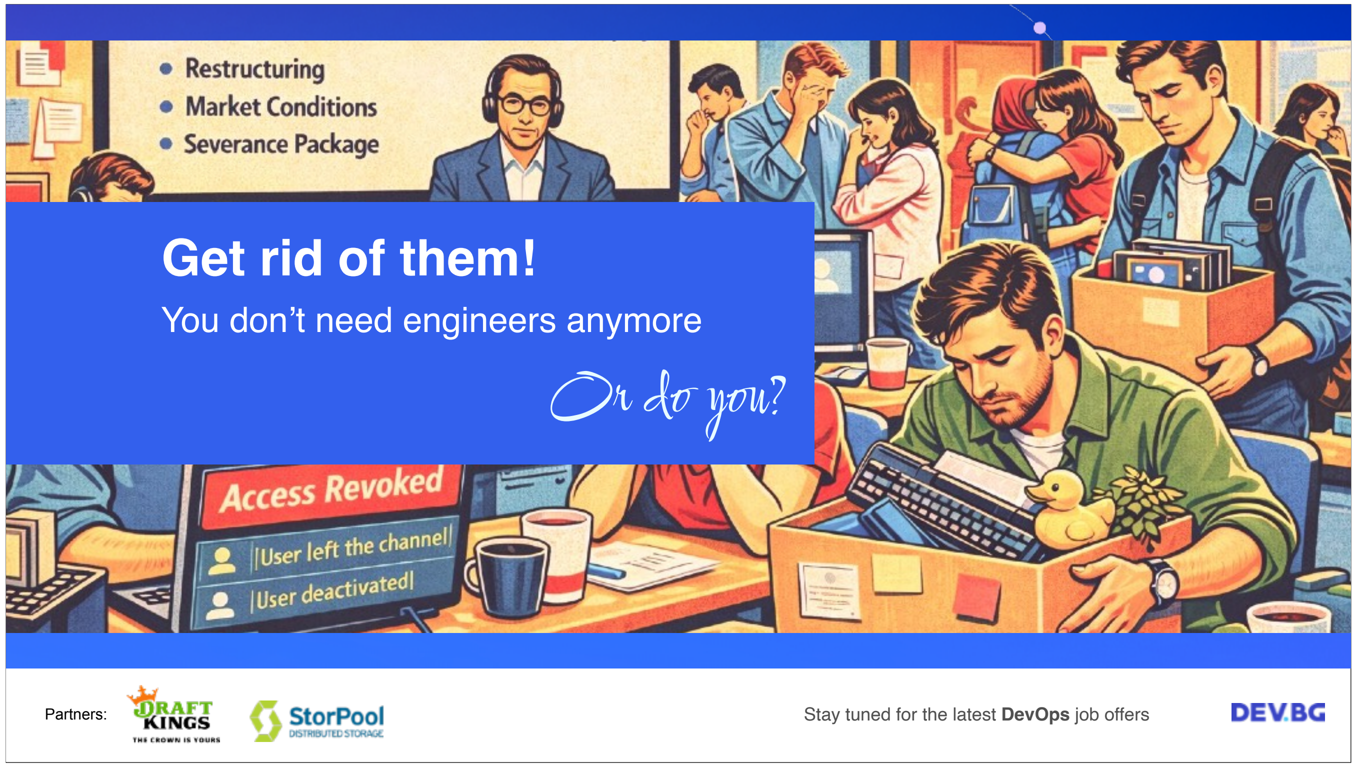
Andrey Makarov
Consultant

I am a leaf on the wind. Watch how I soar.

DataArt



Get rid of them!

You don't need engineers anymore

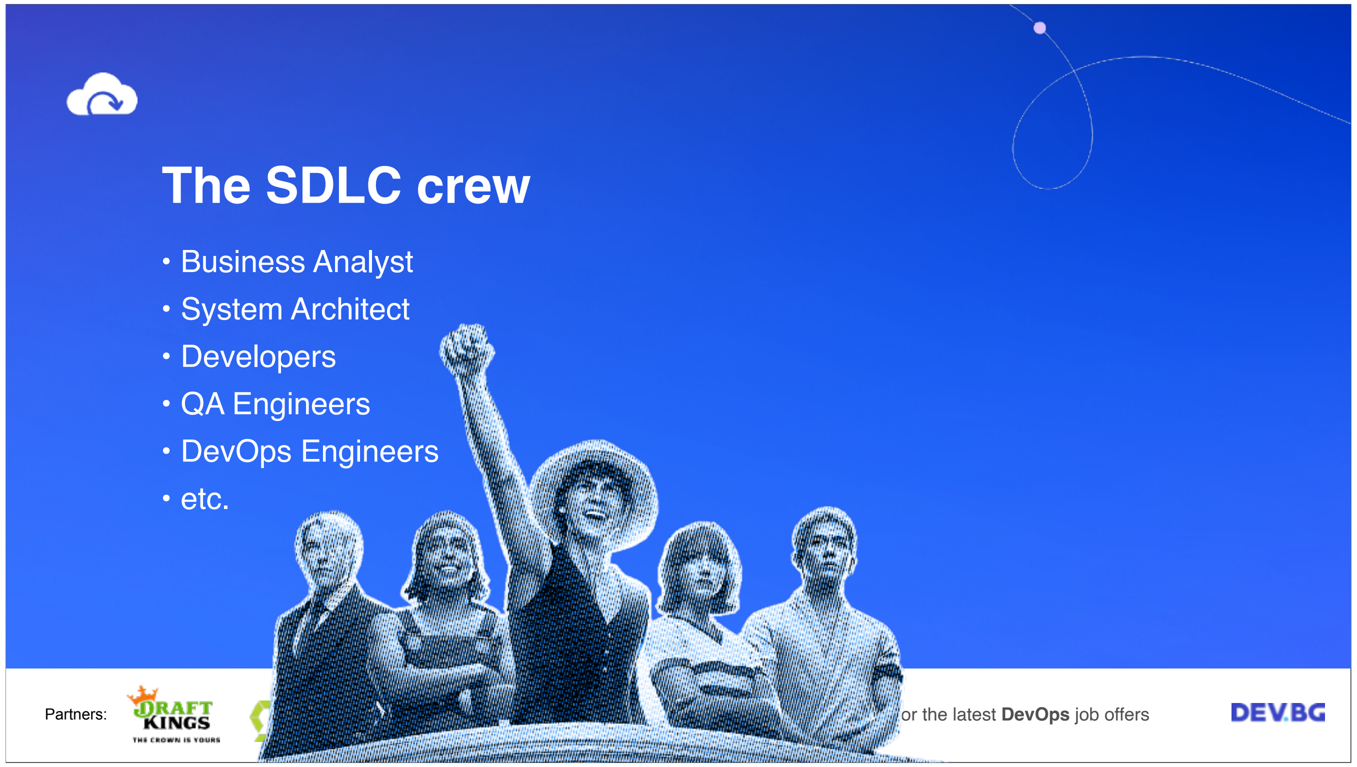
Or do you?

Partners: DraftKings THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers

DEV.BG

- Good evening, everyone! As you can see, my name is Andrey. I've spent almost all my life in Ops, among other fellow software engineers. And today I want to ask you a (slightly) provocative question: What if, as someone in charge, you don't need those fellow engineers anymore? Or... do you? <BROW>
- Stay with me for the next 30 minutes or so, and I promise you'll see something that might change how you think about AI in software development — for better AND (of course, of course!) for worse. Probably in all this seemingly tech-strat talk, there won't be anything that you still don't know, but it's clearly the factor that is outside my zone of responsibility. Alrighy then. Let's dig deeper. <SMILE>



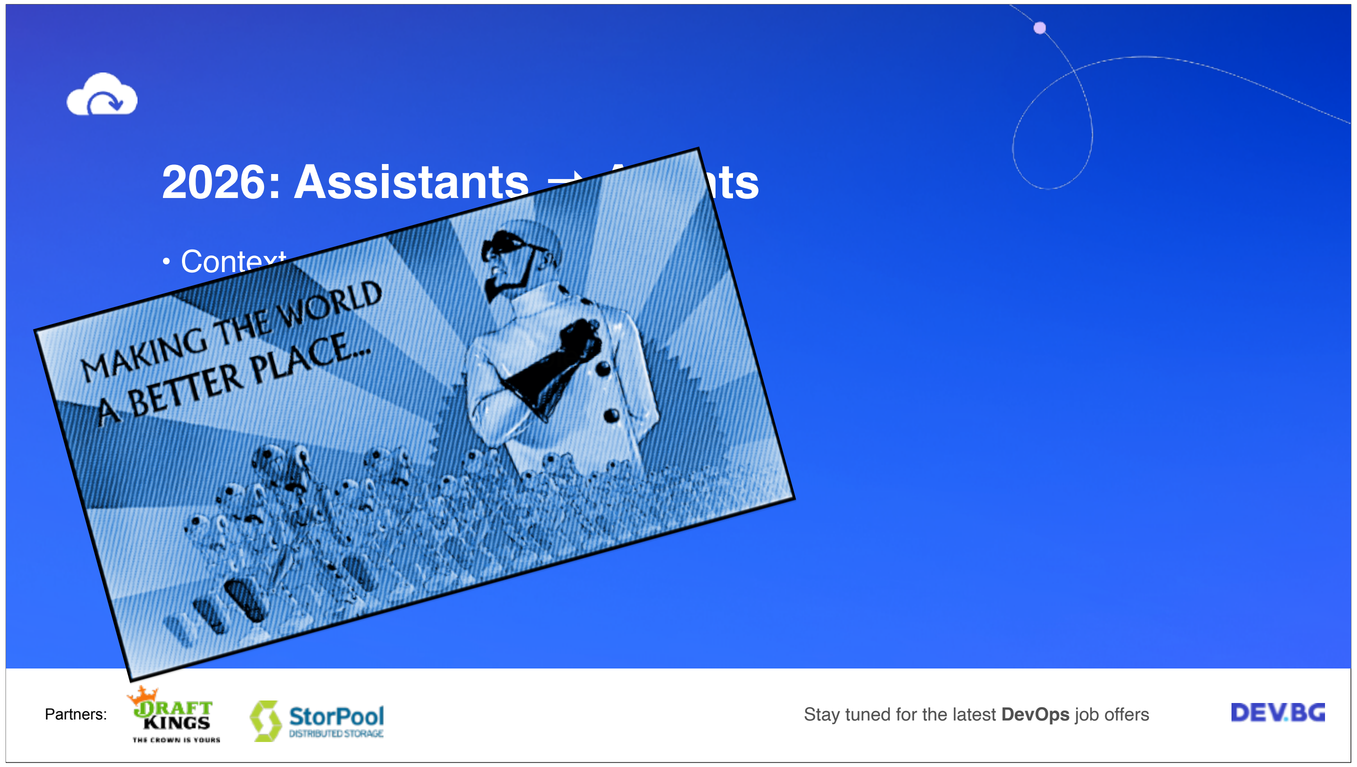
The SDLC crew

- Business Analyst
- System Architect
- Developers
- QA Engineers
- DevOps Engineers
- etc.

Partners:

or the latest **DevOps** job offers

- Let's start with what we know. A traditional software development lifecycle requires a crew (Ahoy!). You need BA to figure out what stakeholders actually want. System architects to design the solution. Developers to write the code. QA engineers to make sure it doesn't break in production. DevOps engineers to actually get it TO production. And that's the simplified version. <TEAM>
- Each role is specialized. Each person spends years mastering their craft. Each one is (very) expensive. Each one has their own very important opinions. Each one can become a bottleneck, outperformer and that (absent) bus driver. <SCEPTICAL>
- So naturally, the question arises: what if we could replace some of them? Or... even <GRIN> all of them?



2026: Assistants → A ts

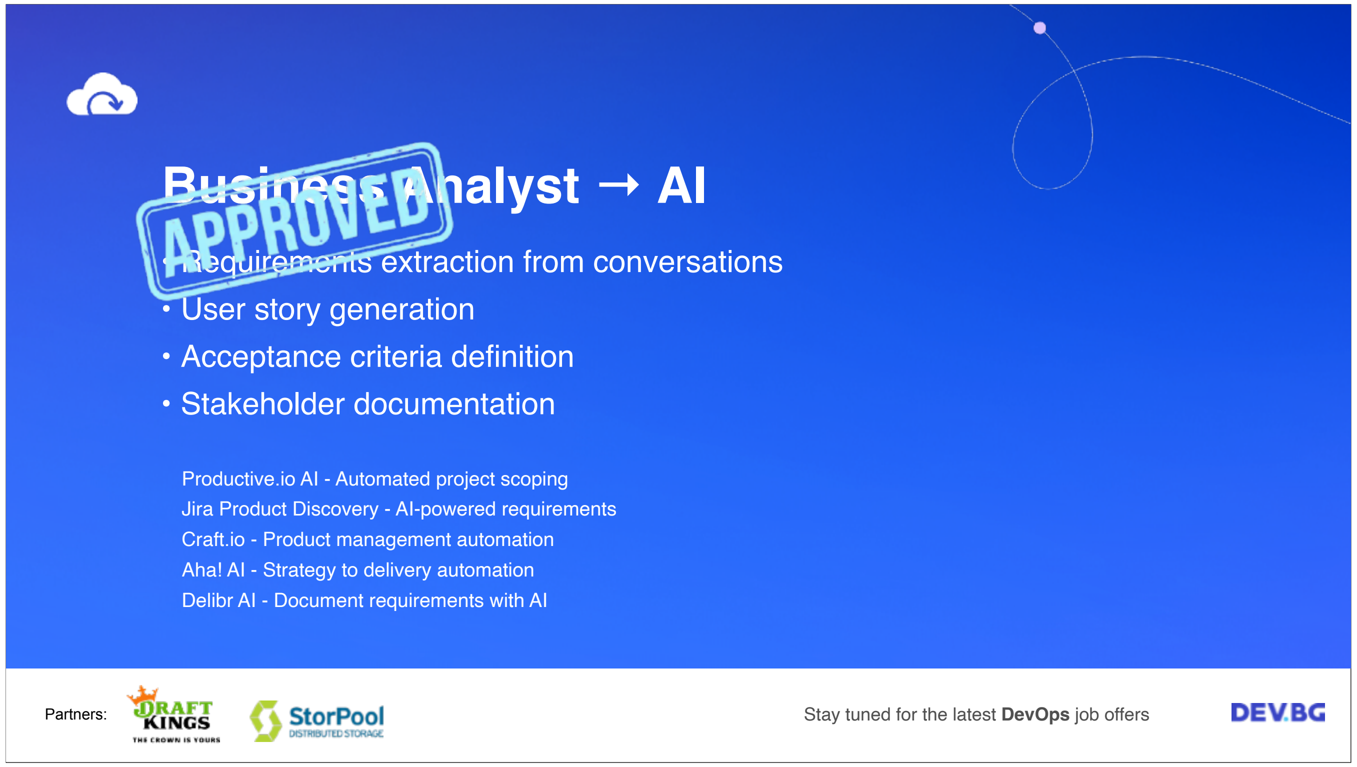
- Context

MAKING THE WORLD
A BETTER PLACE...

Partners:

Stay tuned for the latest **DevOps** job offers

- Here's where we are in 2026. We've moved beyond AI assistants that just autocomplete your code. We're now in the era of autonomous agents. I’m sure you've heard about them. Those little gremlins who can remove your disk D completely. <GRIN> These systems can reason about complex problems, execute multi-step tasks without constant supervision, and possess domain expertise across multiple roles (oh, yes, they are).
- Think about that for a moment. One “AI” system, a few agents, that understands business requirements, system architecture, coding patterns, testing strategies, AND of course so-called DevOps duties – deployment pipelines. Sounds like Christmas! Not theoretically. Today. Right now. <BETTER WORLD>
- This technology definitely has crossed a threshold. Let me show you exactly what I mean. In a small briefing.



APPROVED

Business Analyst → AI

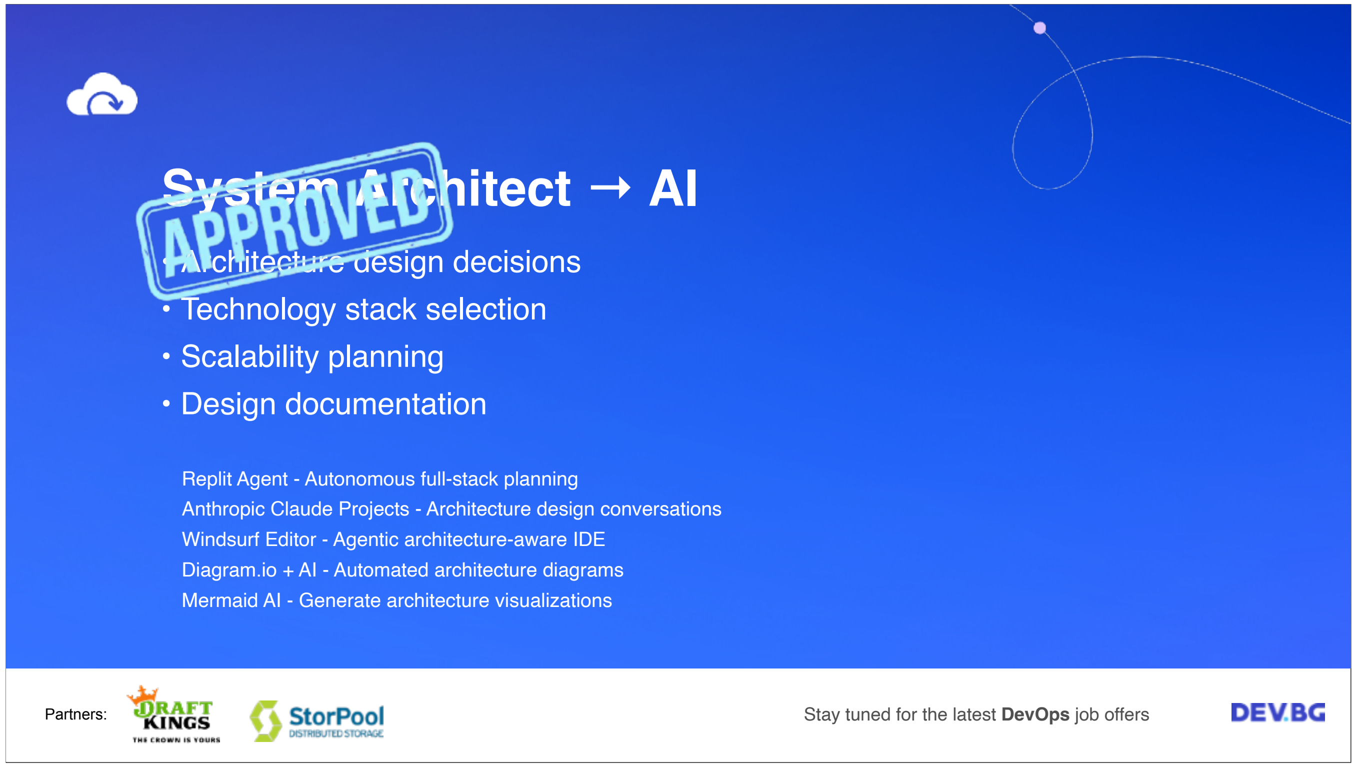
- Requirements extraction from conversations
- User story generation
- Acceptance criteria definition
- Stakeholder documentation

Productive.io AI - Automated project scoping
Jira Product Discovery - AI-powered requirements
Craft.io - Product management automation
Aha! AI - Strategy to delivery automation
Delibr AI - Document requirements with AI

Partners: DRAFT KINGS THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers DEV.BG

- Let's start with the Business Analyst role. Traditionally, BAs spend weeks in meetings, extracting requirements from stakeholders (who don't quite know what they want, oftenly). They document everything in user stories, create acceptance criteria, and produce specification documents.
- Now? Tools like Productive.io and Aha! can automatically extract requirements from very conversations. Jira Product Discovery uses AI to turn vague ideas into structured requirements. Delibr AI generates comprehensive documentation from rough notes.
- <STAMP> The BA's job — translating business needs into technical specifications — can now be done by AI. Not perfectly, but well enough to start.



System Architect → AI

APPROVED

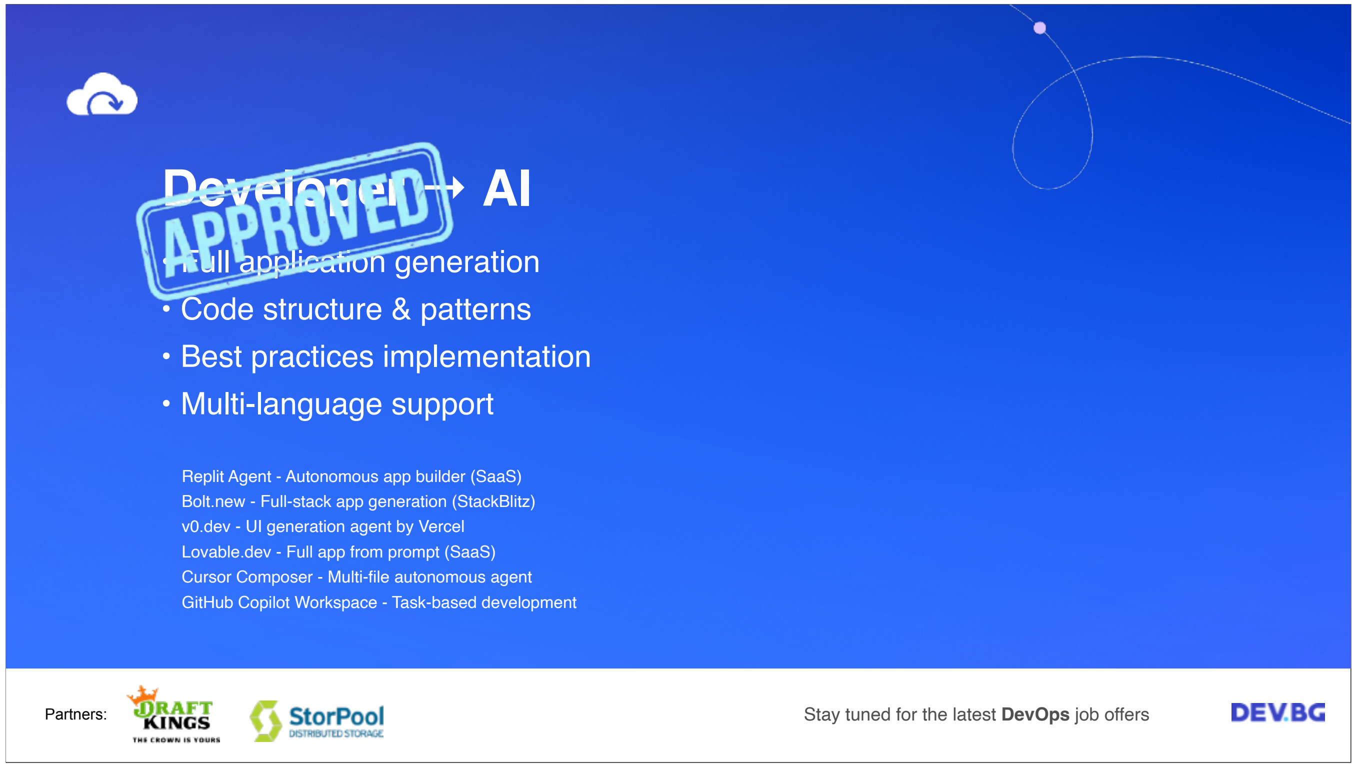
- Architecture design decisions
- Technology stack selection
- Scalability planning
- Design documentation

Replit Agent - Autonomous full-stack planning
Anthropic Claude Projects - Architecture design conversations
Windsurf Editor - Agentic architecture-aware IDE
Diagram.io + AI - Automated architecture diagrams
Mermaid AI - Generate architecture visualizations

Partners: DRAFT KINGS THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers DEV.BG

- Next up: System Architecture. This is supposed to be the senior role, right? Decades (but no more than one and half, or you going to overqualify zone <WINK>) of experience, battle scars from production outages, the wisdom to know which technologies to choose. Or even which _upcoming_ <WINK> technologies to choose.
- But look at what's available now. Replit Agent can plan full-stack architectures autonomously. Claude Projects can have sophisticated architecture design conversations with you (or even with other models), discussing trade-offs and suggesting patterns. Windsurf Editor acts as an agentic IDE that understands architectural context. Mermaid AI can generate architecture diagrams from descriptions.
- <STAMP>The architect's job — making high-level design decisions — is increasingly being automated



Developer → AI

APPROVED

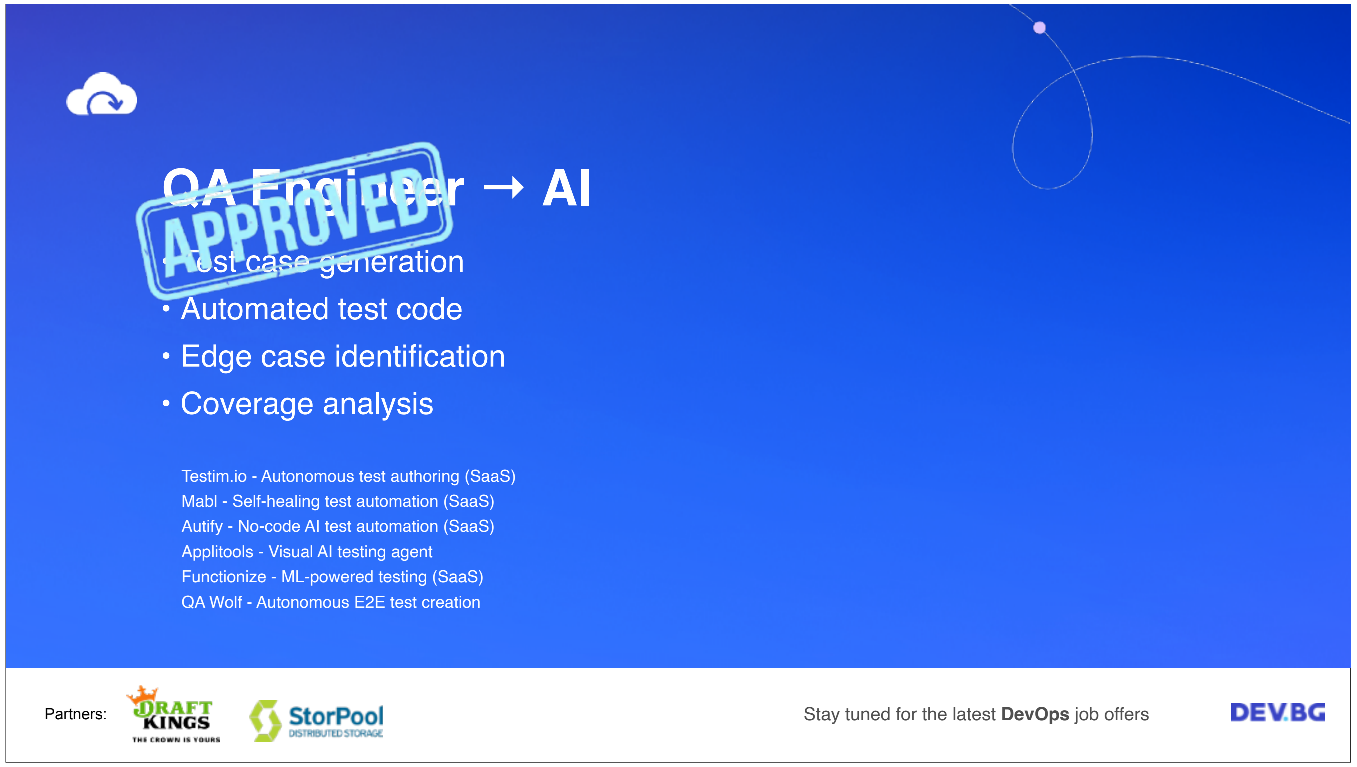
- Full application generation
- Code structure & patterns
- Best practices implementation
- Multi-language support

Replit Agent - Autonomous app builder (SaaS)
Bolt.new - Full-stack app generation (StackBlitz)
v0.dev - UI generation agent by Vercel
Lovable.dev - Full app from prompt (SaaS)
Cursor Composer - Multi-file autonomous agent
GitHub Copilot Workspace - Task-based development

Partners: DRAFT KINGS THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers DEV.BG

- Now we get to the heart of it: Development. This is what most of you in this room do, at least from time to time <SMILE><GIGGLE>, right?
- So… Here's the uncomfortable truth: Replit Agent can build full applications autonomously. Bolt.new generates entire full-stack apps from prompts. v0.dev creates production-ready UI components. Lovable.dev goes from idea to deployed app. Cursor Composer can work across multiple files as an autonomous agent(s).
- And I'm not talking about code snippets. I'm talking about complete applications with proper structure, error handling, and best practices.
- <STAMP> The developer's job — <NOT BAD> translating specifications into working code — is being commoditized before our eyes.



QA Engineer → AI

APPROVED

- Test case generation
- Automated test code
- Edge case identification
- Coverage analysis

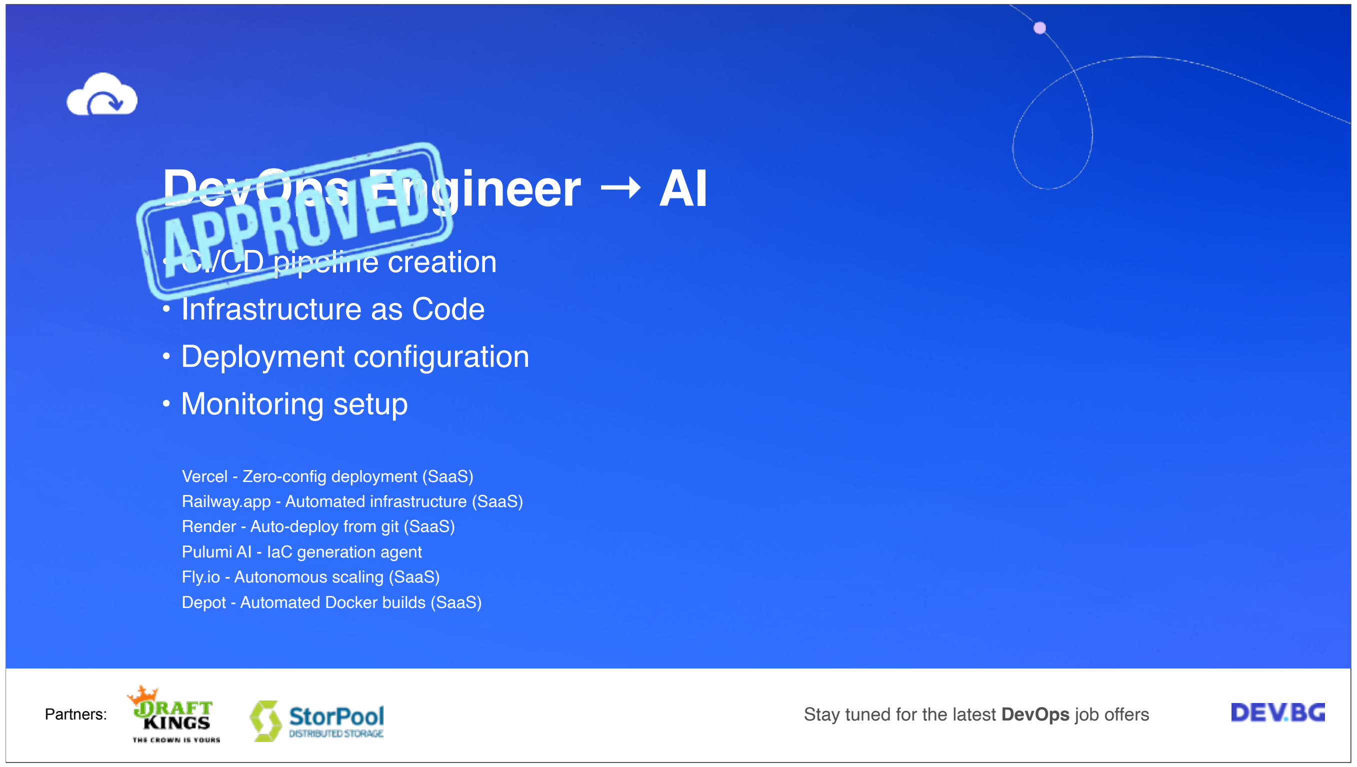
Testim.io - Autonomous test authoring (SaaS)
Mabl - Self-healing test automation (SaaS)
Autify - No-code AI test automation (SaaS)
Applitools - Visual AI testing agent
Functionize - ML-powered testing (SaaS)
QA Wolf - Autonomous E2E test creation

Partners:

Stay tuned for the latest **DevOps** job offers

DEV.BG

- And what about Quality Assurance? <THOUGHTFUL> Hm. Surely we still need humans to think about edge cases and break things creatively?
- Well <SMILE>, Testim.io does autonomous test authoring. Mabl creates self-healing tests that adapt when your UI changes. Autify provides no-code AI test automation. Applitools does visual testing with AI. QA Wolf generates end-to-end tests automatically.
- These systems don't just run tests. They CREATE tests. They identify edge cases you even didn't think of. They adapt when requirements change.
- <STAMP> The QA engineer's job — ensuring quality through comprehensive testing — is being automated.



DevOps Engineer → AI

APPROVED

- CI/CD pipeline creation
- Infrastructure as Code
- Deployment configuration
- Monitoring setup

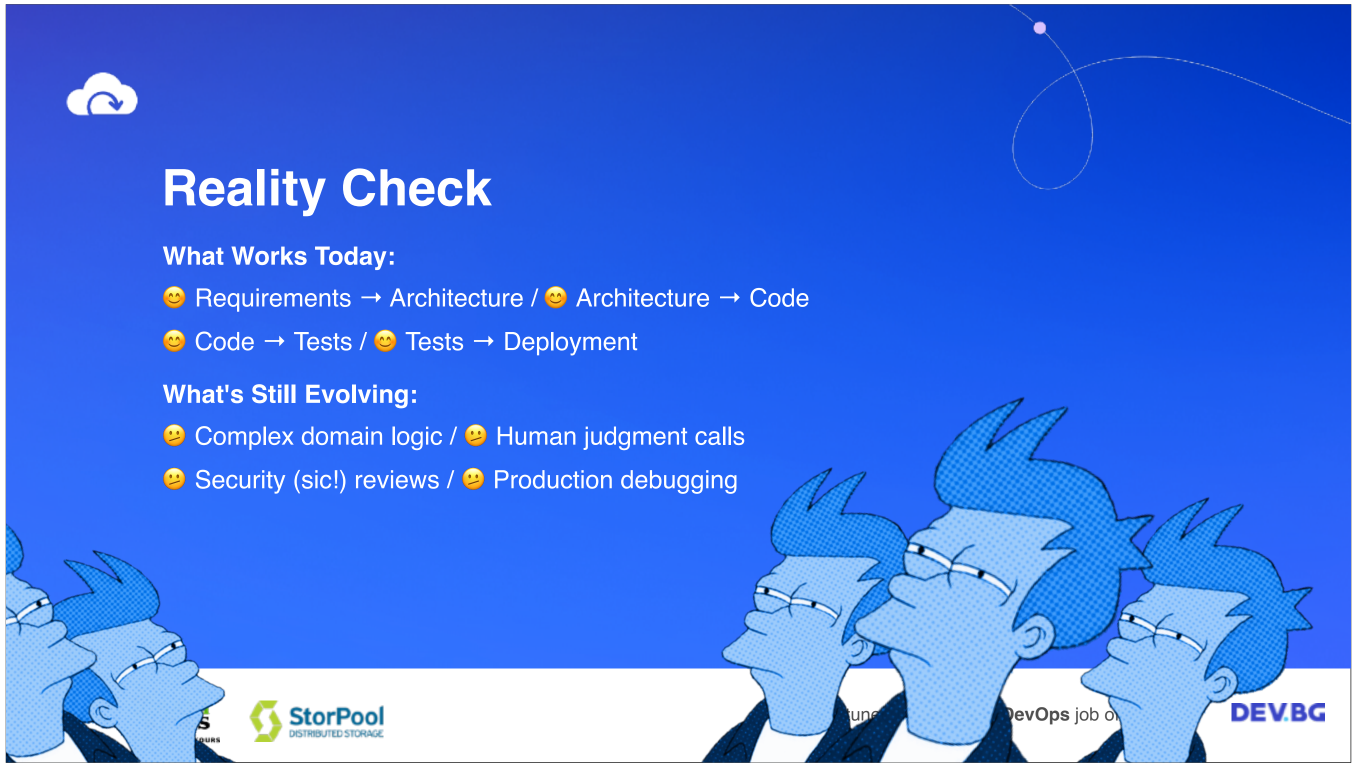
Vercel - Zero-config deployment (SaaS)
Railway.app - Automated infrastructure (SaaS)
Render - Auto-deploy from git (SaaS)
Pulumi AI - IaC generation agent
Fly.io - Autonomous scaling (SaaS)
Depot - Automated Docker builds (SaaS)

Partners:

Stay tuned for the latest **DevOps** job offers

DEV.BG

- Finally, DevOps. This is a DevOps meetup, so let's talk about our jobs.
- Vercel provides zero-config deployment. Railway.app handles infrastructure automatically. Render auto-deploys from git. Fly.io scales autonomously. Depot automates Docker builds. Pulumi AI generates infrastructure-as-code (in many languages).
- Remember when we needed specialized knowledge to set up CI/CD pipeline, manage Kubernetes clusters, and configure monitoring? Now you describe what you want, and it happens.
- <STAMP> The DevOps engineer's job — automating deployment and operations — is itself being automated. Not the first time? Yep.
- **Phew! <SMILE> So, we're done with our short briefing.**



Reality Check

What Works Today:

😊 Requirements → Architecture / 😊 Architecture → Code

😊 Code → Tests / 😊 Tests → Deployment

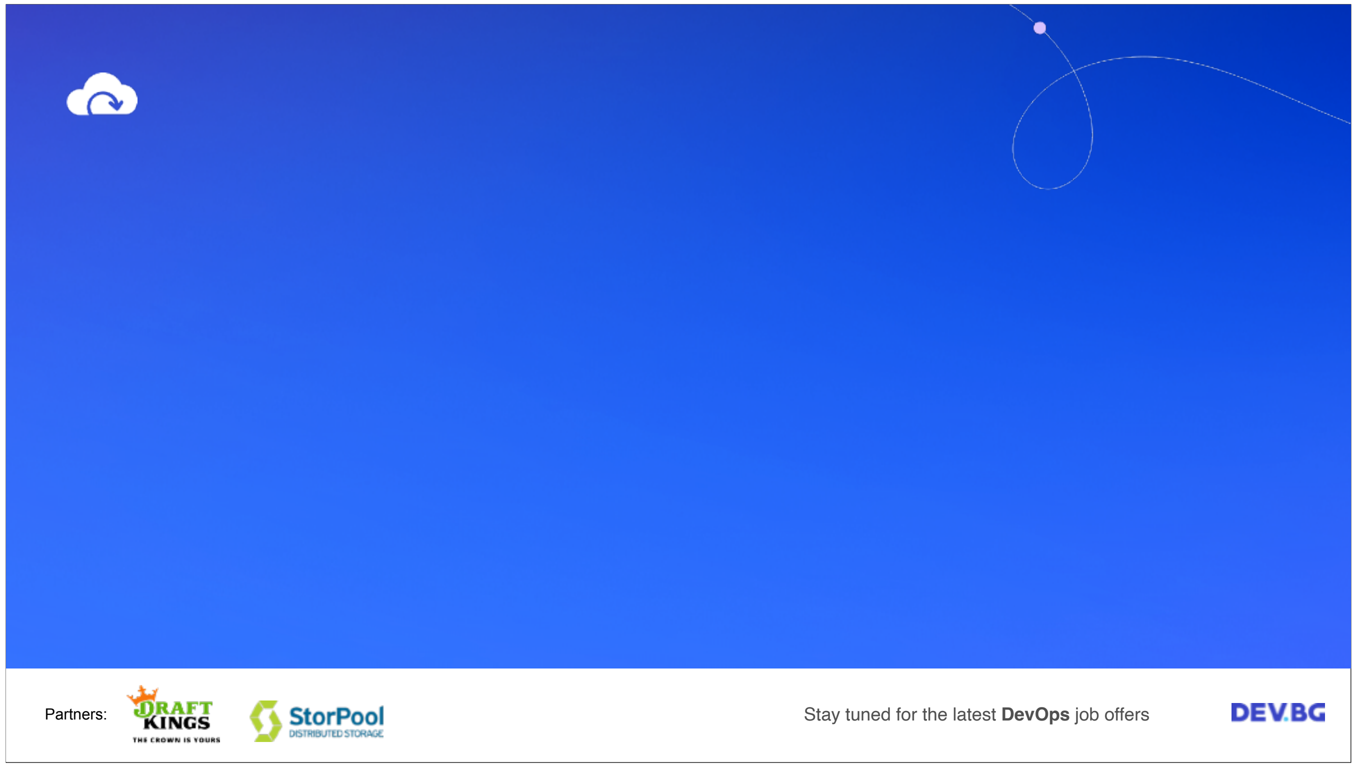
What's Still Evolving:

😕 Complex domain logic / 😕 Human judgment calls

😕 Security (sic!) reviews / 😕 Production debugging

StorPool
DISTRIBUTED STORAGE
DevOps job o
DEV.BG

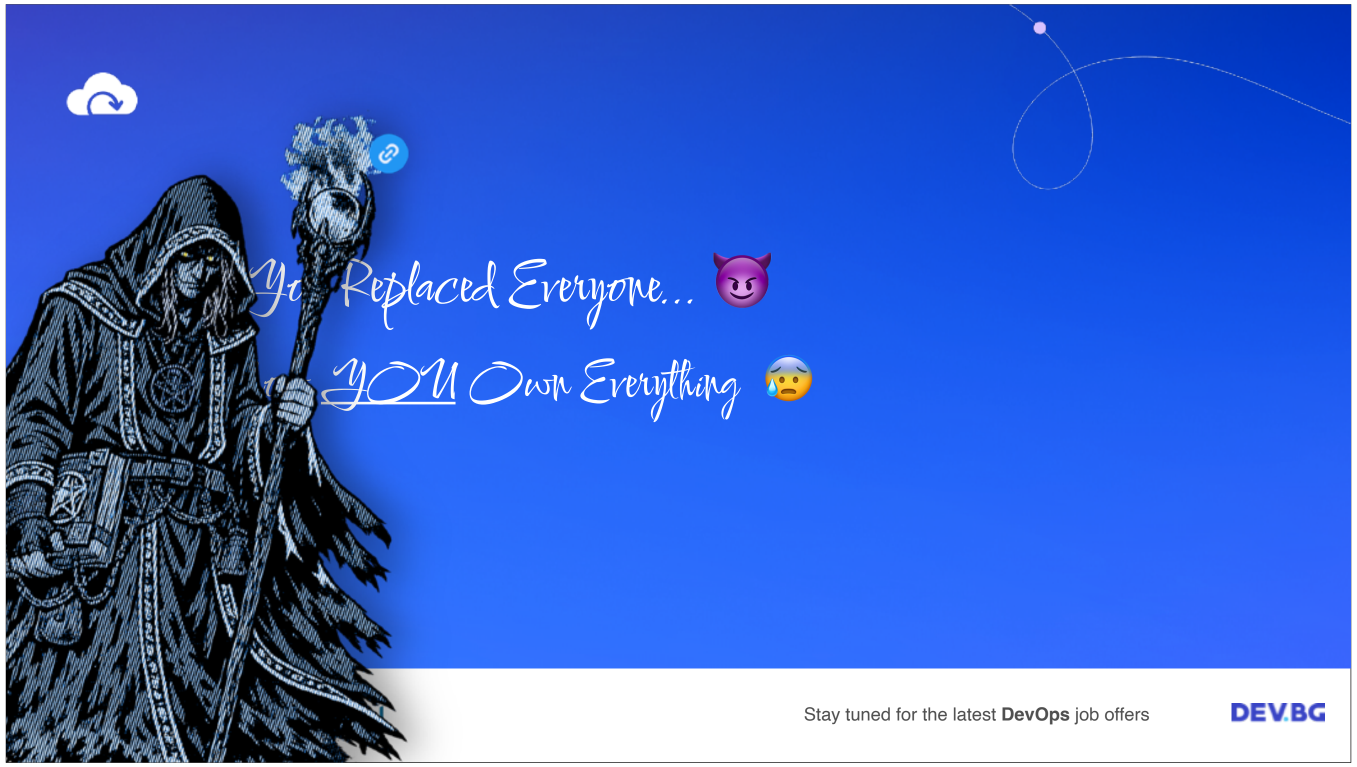
- So let me pause here and do a reality check. What actually works today?
- Going from requirements to architecture? Check. Architecture to code? Check. Code to tests? Check. Tests to deployment? Check.
- The full pipeline exists. The tools work. You can, TODAY, go from a business idea to a deployed, tested application using primarily AI tools.
- Now, what's still evolving? <FRY> Complex domain logic still needs human oversight. Critical judgment calls still need human wisdom. Security reviews still need human <SMILE> paranoia (should our professional skill, I say). Production debugging still needs human intuition.
- But the trajectory is clear. These gaps are closing (right now).



Partners:

Stay tuned for the latest **DevOps** job offers

- So let me show you something from my GitHub. Just a gimmick and mere example. I’m going to do an end-to-end demo right now. We'll start with a simple business idea — nothing fancy, just a real problem that needs solving. And we'll end with a deployed application. All in about 15 minutes.
- I'll be using AI agents, but I'll be orchestrating them with predefined prompts. I'm not claiming full autonomy here — I'm showing you what's possible TODAY when a human guides these tools.
- Before I start: Here no existing code. No pre-recorded magic. This is a small shctuck I'll work through. If something breaks, you'll see it. If the AI makes mistakes, you'll see those too.
- Ready? Let's build something. Three, two, one, let’s go.
- [STARTING LIVE DEMO HERE — 15 minutes]
- Okay. While our little gremlins are working and showing you that you really CAN do the work of multiple engineers with AI assistance, let me tell you why you might not want to. Here's the thing nobody talks about when they're selling you on AI productivity. <SMILE>



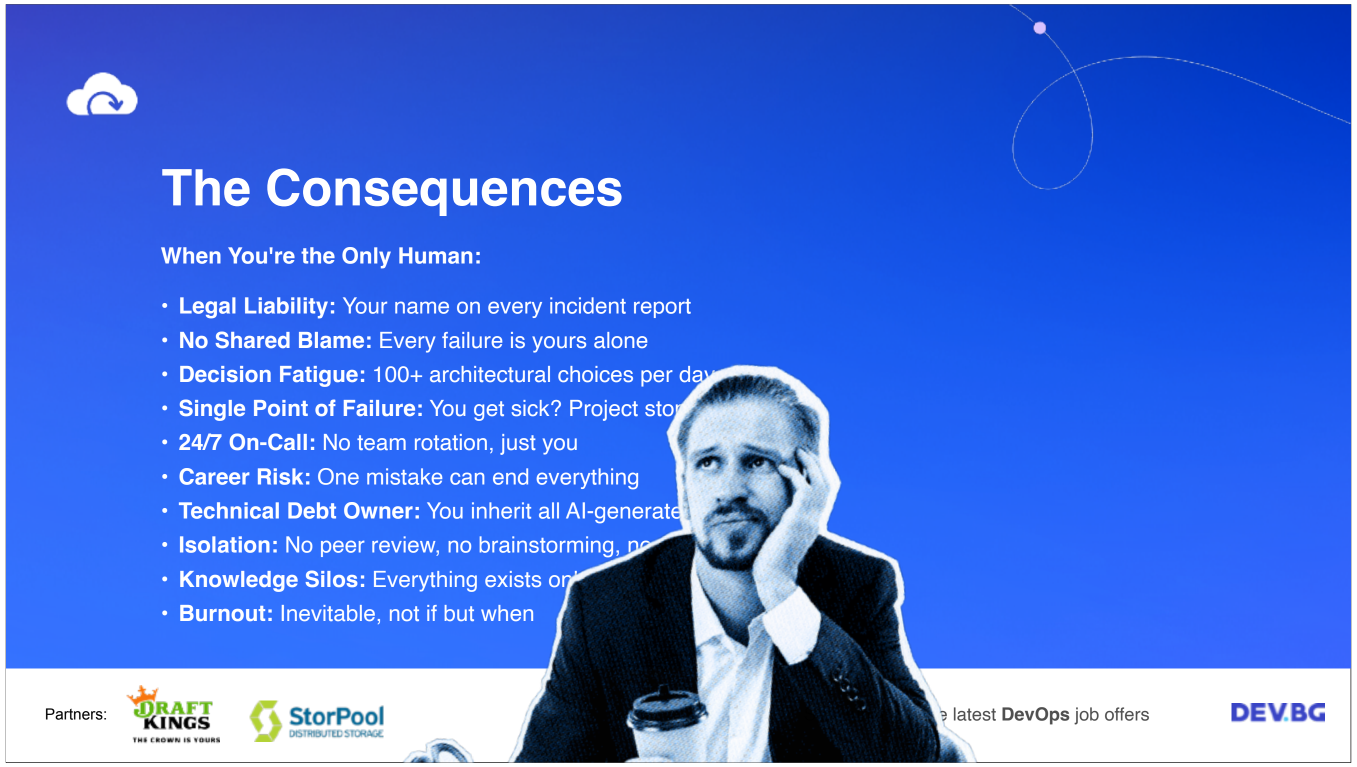
You Replaced Everyone... 😈

YOU Own Everything 😰

Stay tuned for the latest **DevOps** job offers

DEV.BG

- So, when you replace everyone…
- YOU OWN EVERYTHING. And with great power comes great resposibility.
- <PRESS>
- Oh, duck off Reistlin!
- <PRESS>



The Consequences

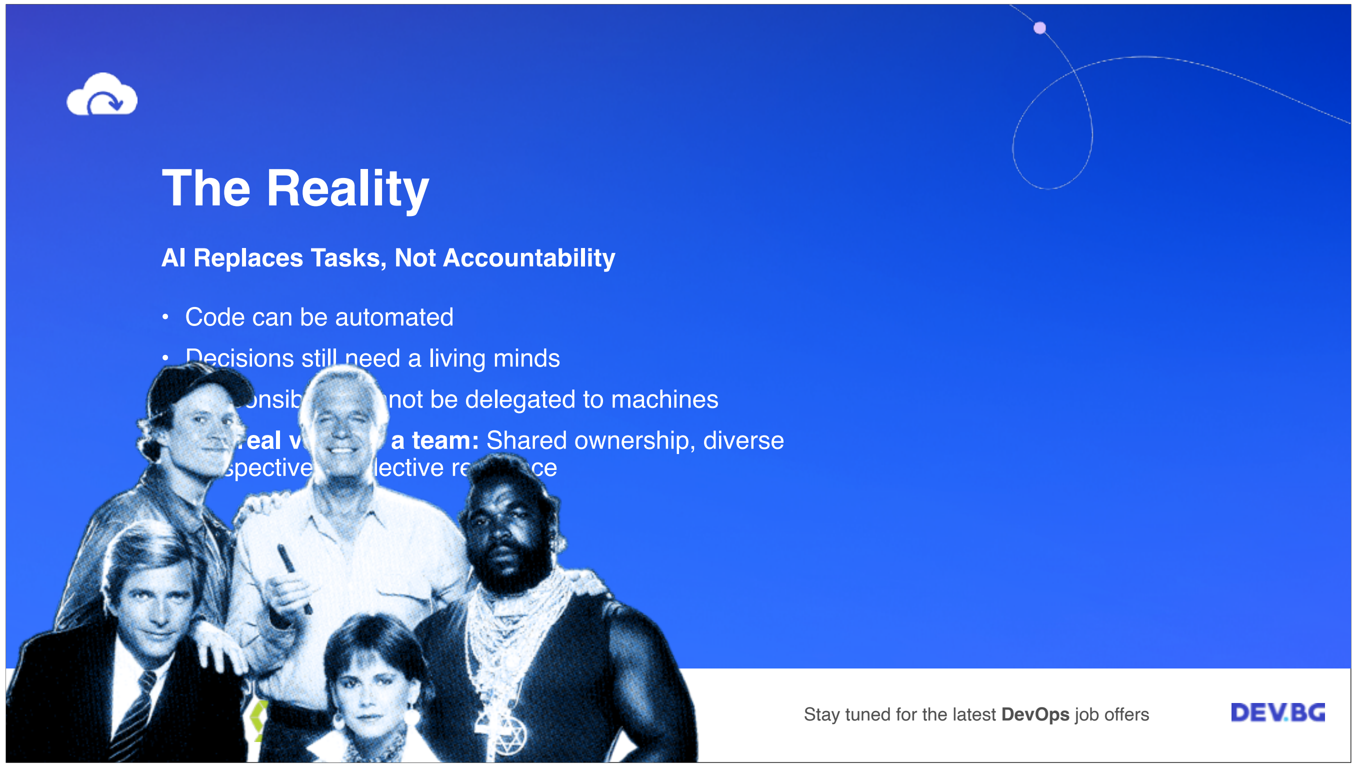
When You're the Only Human:

- **Legal Liability:** Your name on every incident report
- **No Shared Blame:** Every failure is yours alone
- **Decision Fatigue:** 100+ architectural choices per day
- **Single Point of Failure:** You get sick? Project sto
- **24/7 On-Call:** No team rotation, just you
- **Career Risk:** One mistake can end everything
- **Technical Debt Owner:** You inherit all AI-generate
- **Isolation:** No peer review, no brainstorming, no
- **Knowledge Silos:** Everything exists on
- **Burnout:** Inevitable, not if but when

Partners:

e latest **DevOps** job offers

- Think about what happens when you're the only human in the loop.
- Legal liability? Every incident report and every judicial claim has your name on it. There's no "the team made this decision." It's just you.
- Something fails? No shared blame. No collective responsibility. Every failure is yours alone.
- Decision fatigue? You're making 100+ architectural choices per day. Choose the wrong database? Your fault. Pick the wrong framework? Your fault. Every. Single. Decision.
- You get sick? The project stops. You're the single point of failure. No coverage. No backup. Just you.
- On-call rotation? What rotation? It's 24/7, and you're the only one on the roster.
- Career risk? One major mistake, one security breach, one production outage — and there's no team to distribute the consequences. It all lands on you.
- Technical debt? All those shortcuts the AI took to move fast? You inherit them. You maintain them. Forever.
- Isolation? No peer review. No brainstorming sessions. No "hey, can you take a look at this?" You're alone with your decisions.
- Knowledge silos? Everything exists only in your head. The AI doesn't remember context between sessions. You're the sole keeper of institutional knowledge.
- Burnout? Not "if." WHEN. This isn't sustainable. Human beings aren't designed to carry this much cognitive and emotional load alone.



The Reality

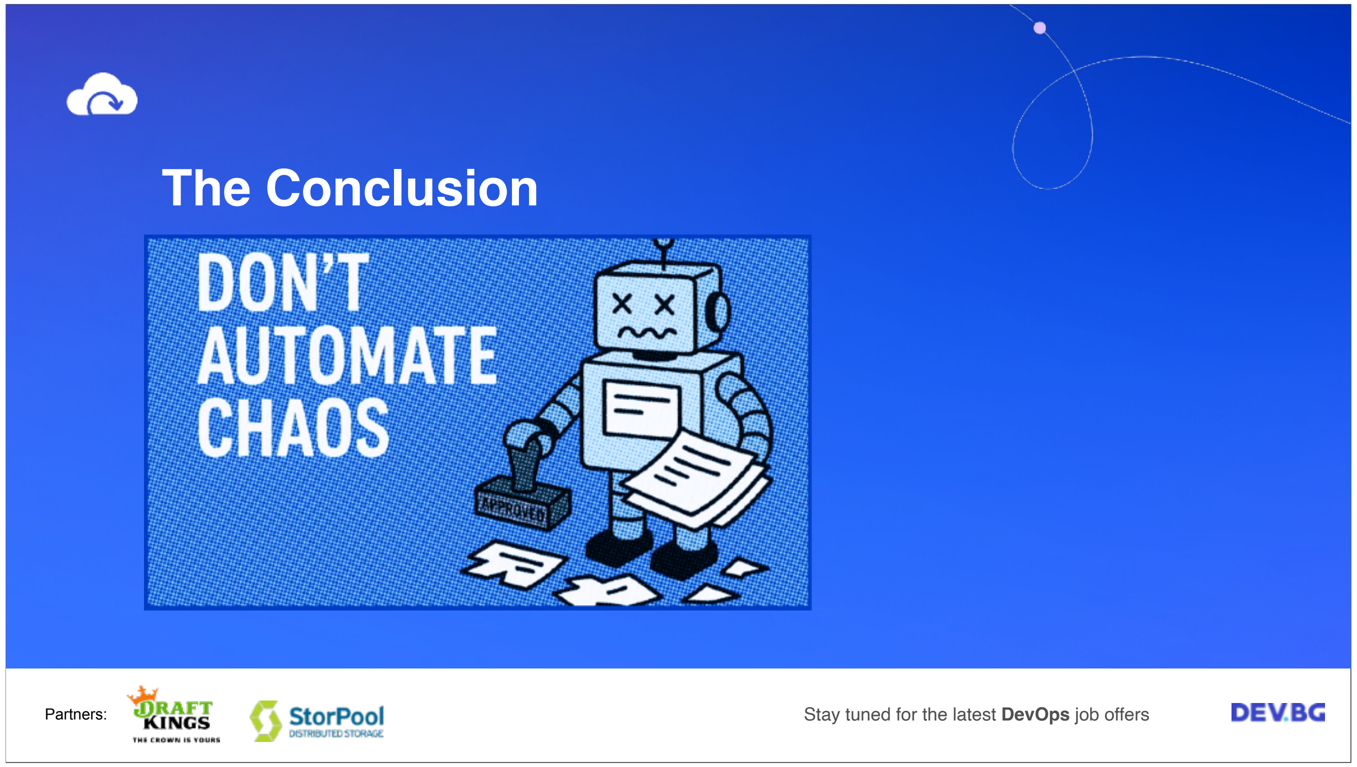
AI Replaces Tasks, Not Accountability

- Code can be automated
- Decisions still need a living minds
- Responsibility cannot be delegated to machines
- **The real value of a team:** Shared ownership, diverse perspectives, collective resilience

Stay tuned for the latest **DevOps** job offers

DEV.BG

- Here's what I've learned: AI can replace tasks. AI cannot replace accountability.
- You can automate code generation. You can't automate the responsibility for that code.
- You can automate deployment. You can't automate the decision of WHETHER to deploy.
- You can automate testing. You can't automate the judgment of whether you've tested enough.
- Responsibility cannot be delegated to machines. And that's actually the real value of a team — it's not just about getting work done. It's about shared ownership, diverse perspectives, and collective resilience.
- A team can survive mistakes. A team can challenge bad decisions. A team can cover for each other. A team can learn together. <PRESS>
- One person with AI? You're efficient until you're not. You're fast until you break. You're productive until you burn out.

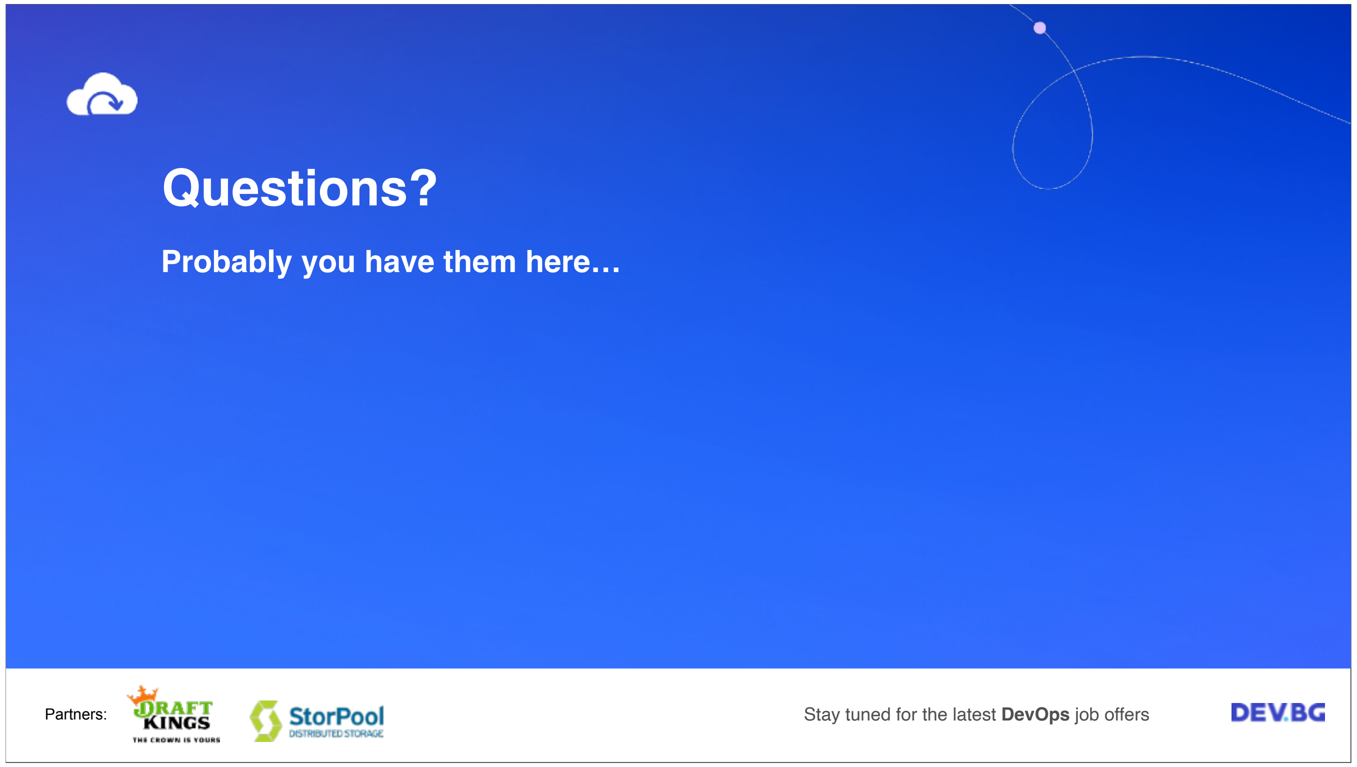


The Conclusion

Partners: DRAFT KINGS THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers DEV.BG

- So, can AI replace your dev team?
- Technically? We're getting close. The tools exist. The capabilities are real. What I showed you tonight isn't science fiction.
- Practically? You'll wish it couldn't. Because the human cost of being the only person responsible for everything is too high.
- The future isn't about replacing teams with AI. The future is about using AI as a force multiplier FOR teams. Let AI handle the repetitive tasks, the boilerplate code, the routine configurations. Use that time for what humans are actually good at: creative problem-solving, strategic thinking, collaboration, and sharing the weight of responsibility.
- The wisdom here isn't "can we replace engineers?" The wisdom is: empower your engineers with AI, don't eliminate them.
- Because at the end of the day, software isn't just code. It's decisions, trade-offs, and accountability. And those things? The busyness stuff? Those still need humans. And their responsibility. So, if you in charge, just think about it on the next wave of AI-driven buzz before you'll cut another job.
- How long will we be in this balanced state? Who knows? Keep going, read the news. Should we be afraid of the future? Of course, as usual, it's human nature. Was the content of this barely-surface speech completely generated by LLM. May be. But it's my responsibility. <SMILE>



Questions?

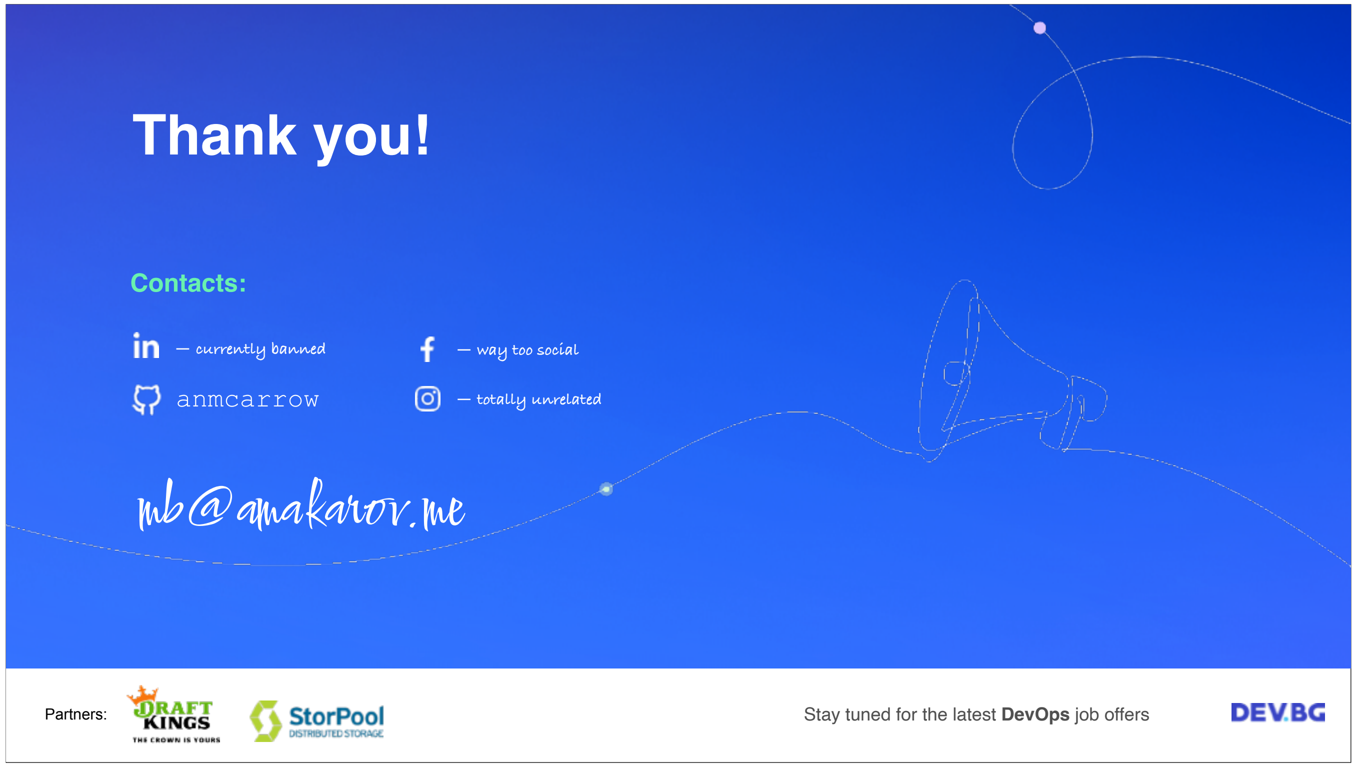
Probably you have them here…

Partners: DRAFT KINGS THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers

DEV.BG

- Let’s see what do we have in chat…
- And let’s provide the favor of answering to our little gremlins itself <SWITCH TO GPT>



Thank you!

Contacts:

in — currently banned

— way too social

anmcarrow

— totally unrelated

mb@amakarov.me

Partners: DRAFT KINGS THE CROWN IS YOURS StorPool DISTRIBUTED STORAGE

Stay tuned for the latest **DevOps** job offers

DEV.BG

- Thank you, everyone. It was fun and a pleasure. And your questions was gorgeous. Thank you. <SMILE> So, now let's get back to work as the good engies we are. And you can get all those posh slides plus demo files from my GitHub account. You are welcome to play with them. They're all yours.

OST Info

Fahrstuhlmusik Muzak & Jazz by Parodoro Beats

Link: youtu.be/hHaz2-cZ_1A

License infos:

- Kevin MacLeod (CC BY 3.0): https://creativecommons.org/licenses/by/3.0/
- Bensound (Freie Lizenz): https://www.bensound.com/licensing
- Congusbongus, Glitch Game, Joth [CC0]/Public Domain: https://creativecommons.org/publicdomain/z...
- Migfus20, Stonefree [CC BY 4.0]: https://creativecommons.org/licenses/by/4….
- Strobotone (CC BY-ND 4.0): https://creativecommons.org/licenses/by-nd…
- Separate Usage infos + Social Media Links of all artists: https://bit.ly/Links-2
- Archive links with License as evidence that this music released under a free License: https://bit.ly/Archivlinks-Loungemix