digit

Powered by sTARTUp Day

Digit 2026

Driving the Change

9 OCTOBER 2026 AT KAMMIVABRIK IN TARTU

EVENT SUMMARY

digit
Powered by sTARTUp Day



Looking back at Digit 2026, what makes me proud is the community we brought together. Seeing so many smart and passionate people sharing ideas and learning from each other was a reminder that Driving the Change is all about curiosity, collaboration, and the willingness to challenge our perspectives.

A huge thank you to our speakers, partners, volunteers, and the entire organising team for bringing Digit to life. And to everyone who joined us - thank you for coming and for being part of this community!

- Birgit Jürgenson
Event Lead

It seems that we are still in the early stages of adapting our processes to this new technology, but it also seems that we all agree on one thing: the bottleneck has moved away from writing code. Some trust AI fully and some are more skeptical, but we all seem to agree that things have changed. I hope tech leads and developers found inspiration to learn and to adapt their existing development cycle to this new environment. All in all, adapting to and keeping up with new technology has a lot to do with leaders: how much time do you dedicate to learning new things? Right now it feels like more should be dedicated, because the ROI on this investment is too big to ignore.

- Siim Kallari
Program Manager

The idea I keep coming back to from Digit 2026 is that many of the paths to success with AI turned out to be surprisingly old-fashioned: writing good documentation, but for AI. Whether skills and role definitions for agents, AGENTS.md files in a repository, or a semantic layer over a data warehouse, the teams getting the most out of LLMs were the ones that had done the work of writing down what they know and what they want. Around that, the range of practice was striking, from cautious first steps with vibe coding, sometimes deliberately limited out of longer-term concerns, to organizations letting agents operate fully autonomously in parts of their business. Several years into commercial chatbots and agentic coding, how and when to use these tools is far from settled. What I hope people took home is that the unglamorous groundwork, clearly articulating how your own systems and data work, pays off whichever way that question resolves.

- Andreas Beger
Data & ML Track Manager

DIGIT IN NUMBERS

digit
Powered by sTARTUp Day

3
stages

8
workshops

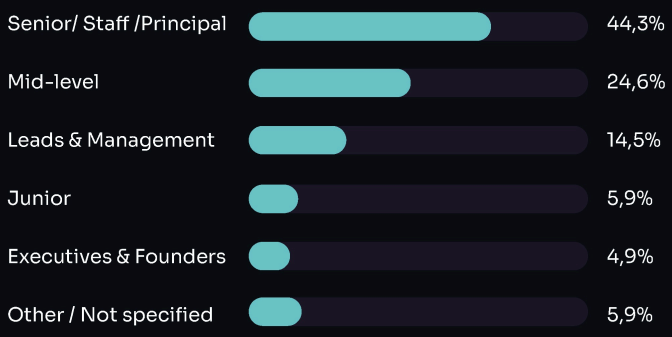
611
participants

37
speakers

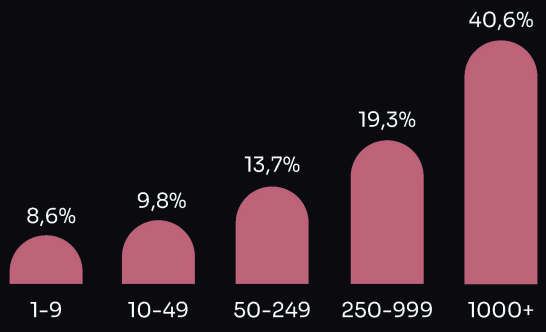
126
companies & organisations

16
countries represented

EXPERIENCE LEVEL



COMPANY SIZE



16
COUNTRIES REPRESENTED

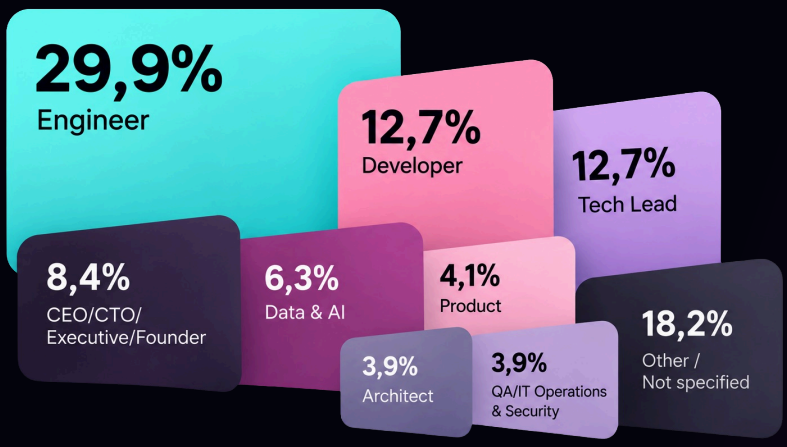
Bulgaria
Canada
China
Costa Rica
Czechia
Estonia
Finland
Italy
Latvia
Poland
Portugal
Romania
Spain
Switzerland
Ukraine
United Kingdom

PARTICIPANTS BY ROLE

Software engineers formed the largest group, alongside technical leaders, founders, and specialists.

The audience also included executives, founders, and specialists across data, AI, product, architecture, and IT operations.

*Based on job titles provided at registration.



digit
Powered by sTARTUp Day



SPEAKERS

Andrii Kravets
Founder @ Small Red Flowers

Agur Jõgi
Tech Enthusiast, former Pipedrive CTO

Javier Ortin Cervera
Web Developer @ Easy Feedback

Mauro Pelucchi
Vice President Data Science @ Lightcast

Siim Vene
Group Chief Architect @ Piletilevi PLG

Christian Heitzmann
Owner @ SimplexaCode

Andrei Maier
Chief Architect @ SEB

Stenver Jerkku
CEO & Co-Founder @ Soldera

Vitalii Lakusta
VP of Engineering @ Pactum AI

Mart Parve
Head of Software Engineering @ Hevi Optronics

Kristiina Kapten
Solution Architect @ SEB

Elis Laasik
UX/UI Designer and Developer @ Codeborne

Natia Kukhilava
Co-Founder & CTO @ Helio AI

Immanuel Sanka
Bioinformatics Platform Engineer @ Icosagen Cell Factory

Karmen Tigas
Talent Manager @ //kood

Ivan Elorza Tahum
Co-founder @ Amiqa Tech AB

Kevin Sutanto
Analytics Engineering Manager @ Bolt

Priit Kaasik
Human Founder @ nodachi.ai

SPEAKERS

digit
Powered by sTARTUp Day

Ivo Kund
Co-Founder & CTO @ Closaria AI

Carlos Trujillo
Principal Data Scientist @ PyMC Labs

André Simões
Software Architect @ TEKEVER

Tuule Sõber
Engineering Manager @ Zendesk

Juri Gavshin
Principal Software Engineer @ Bolt

Klaus-Eduard Runnel
Software Architect @ Cybernetica

Anastasija Nikiforova
Associate Professor of Applied AI and Information Systems @ University of Tartu

Yaroslava Mykhailenko
Data Scientist @ Enefit

Kirill Linnik
Principal Software Engineering Manager @ Microsoft

Evgenii Indenbom
Principal Machine Learning Scientist @ Microsoft

Lauri Võsandi
Founder @ Codemowers

Erki Aas
Senior Product Engineer @ Baltic LEI

Alexandru Neagu
Cloud Engineering Director @ Oracle

Hendry Sadrak
Co-Founder & CTO @ Modash

Karel Jakubec
Customer Engineer @ Google

Ain Arend
Co-Founder & CTO @ 99math

Iya Rõõs
Data Scientist @ Wise

Iremsu Savas
Machine Learning Engineer @ Pipedrive

Heili Aavola
Junior Research Fellow in Machine Learning @ Institute of Computer Science, University of Tartu



Why choose Tartu?

Tartu is a vibrant, youthful and friendly city located in Southern Estonia. With a population of ~100 000, it is the second biggest city in Estonia, but globally, we consider it the **world's biggest small town**.

Considering moving to Tartu? **The city welcomes you with open arms!**

More life during one lifetime



Tartu is a safe and compact 15-minute city, where you can easily get everywhere by cycling, walking or using public transportation.



There are several green areas, boulevards, benches, public tap water spots and public WCs.



The city is family friendly – you'll find schools, kindergartens, higher education; hobby groups and cultural events for all age groups.



The schools offer warm, healthy, organic meals at schools for less than 2€ per pupil.

The vibe of Tartu is the best therapist



Tartu is planned according to the principle that the living environment affects mental health.



Public outdoor gyms, running and skiing trails, as well as swimming spots? All present in Tartu!



Several public cultural events are organized in different parts of the city all year around.



Movement-supporting free events for both the youth and the elderly.

The hub of happenings



Something is always happening in Tartu! Concerts, festivals, sports events... you name it.



The city is home to Estonian national theatre, and the national museum.



It's time to sTARTUp! Tartu is an ideal test environment for cutting-edge technologies of the future.



Strong sectors in Tartu: ICT, wood and metal, health-tech, food industry, film, creativity

Varia



Around 20% of the city's population are students.



Tartu is comfortably connected to the world by the air, the railway and the road.



Estonia has 4 seasons. Both snowy winters and long summer nights are a reality here.



Tartu Welcome Centre offers free consultations, services and events for newcomers in the city and nearby.

TARTU

Contact us! **Tartu Welcome Centre**

info@tartuwelcomecentre.ee
Facebook

digit
Powered by sTARTUp Day

Key ideas from our speakers

SWE STAGE

Software engineering in the age of LLMs

Technical talks on integrating LLMs into real systems: architecture, scalability, reliability, and trade-offs in practice. By developers, for developers.

Data & ML STAGE

Python, ML, and AI in depth

Ecosystem-specific talks, deeper dives into tooling, and discussions on where Python is heading.

LLM Overflow Stage

Practical discussions and hands-on demos on building and working with LLMs and agents.

Workshops

Hands-on sessions on LLM integration, developer workflows, and staying relevant.





SWE stage – Software engineering in the age of LLMs

Your LLM Is Not Your Moat: Seeking Alpha and Staying Relevant in the Age of AI

Andrii Kravets
Founder @ Small Red Flowers

- Access to AI is not a moat. When everyone can use the same models, your advantage comes from how you apply them and the value you create.
- Your unique combination of skills is your edge. Your experience, taste, and judgment give your work direction and personality. Use AI to amplify what makes you distinctive.
- Amplify your strengths. Challenge your blind spots. AI can reinforce your mistakes too. Question its advice most carefully in the areas you understand least.
- Act, experiment, and persist. Nobody knows the future. Take calculated risks, test ideas against reality, and keep learning through setbacks. Agency and persistence help turn shared tools into an advantage.

Skills no one talks about, but everyone hires for

Karmen Tigas
Talent Manager @ //kood

- Companies value engineers who combine deep technical expertise with product thinking and communication skills, delegate execution rather than thinking to AI, and stay curious.
- The goal is not simply to build software, but to solve the right problem and understand what improved as a result.

SWE stage

digit
Powered by sTARTUp Day

MLVC: Building the World's Most Efficient Video Codec

Evgenii Indenbom
Principal Machine Learning Scientist @ Microsoft

- Compression efficiency is only part of the challenge. Neural video codecs also need to overcome computational costs and cross-platform compatibility issues to work outside the lab.
- MLVC brings neural video compression to commodity hardware. The program describes up to 6× lower bitrate than H.265, deployment in Microsoft Teams, and an open-source release.
- Making ML-based compression practical requires engineering trade-offs. The session explores what it takes to deploy a neural video codec at scale.

The Life Agentic with Priit & Claude

Priit Kaasik
Human Founder @ nodachi.ai

- AI-assisted is an upgrade to the old job. AI-native is a different job — the difference is who does what, not which tools you use.
- First We Prompt. It cannot work otherwise. A team that keeps a manual escape hatch never crosses over.
- The Life Agentic is a loop — adventure, then check. Judgment and direction start the adventure; taste and verification run the check. Skip the check and you're not AI-native, you're just fast.
- One human plus agent formations is a real operating unit today. Nodachi was built from zero this way — this talk is an artifact of the model.

Artificial Intelligence adoption in Enterprise environment

Andrei Maier
Chief Architect @ SEB

- Artificial intelligence alone is not enough.
- Context is the key to everything.
- AI adoption scaling in Enterprise.

When AI Enters the Product: What Happens Between a Working Model and a Working Product?

Iremsu Savas
Machine Learning Engineer @ Pipedrive

- A working model isn't a working product. Most of the hard problems live around the model, not inside it.
- Define "better" before you pick a model. Benchmarks narrow the search; your product decides.
- Where and when AI runs is a design decision. Put it in the user's workflow, and question whether every inference needs to happen at all.
- Measure acceptance, not just accuracy. And before optimising the same thing again, ask: is this still the bottleneck?

Learning to Code with AI: What It Means for the Next Junior Developers

Javier Ortin Cervera
Web Developer @ Easy Feedback

- Students (potential junior Devs) can't really tell what they know and don't know.
- Also they understand that AI might make them lazy and that they still need to put the hard work to really learn something.
- It's also very positive that they are eager to learn and put the hard work. They might just need an experienced hand in choosing the learning path. So they need feedback on what are the gaps they need to learn or fix.
- One good way to help them learn is by giving them broken systems where the connections are missing.

SWE stage

digit
Powered by sTARTUp Day

Engineering from Zero to Hero How AI Is Accelerating Modern Engineering Cycles

André Simões
Software Architect @ TEKEVER

- AI changes how engineering happens, not just how code is written. At TEKEVER, this means redesigning parts of a traditional engineering workflow.
- Reducing repetitive work can accelerate delivery. The session explores where AI helps teams build faster within a complex, real-world operation, and where it falls short.
- The biggest shift is in the process. Getting value from AI requires examining how engineering work is organized, rather than focusing on the technology alone.

The Future of Agentic Engineering — When Capability Isn't the Bottleneck Anymore

Siim Vene
Group Chief Architect @ Piletilevi PLG

- The bottleneck is shifting from AI capability to organizational discipline. The talk argues that models have crossed a capability threshold, making how teams organize their work increasingly important.
- Mature systems carry institutional memory in their code. Agentic engineering needs to account for the knowledge and decisions embedded in existing systems.
- Engineering roles are moving from execution toward orchestration. The session explored what this shift means in practice and offered a route beyond the hype.





Data & ML Stage – Python, ML, and AI in depth

Synthetic Counterfactuals for Text Classification: Detecting AI-Written Job Postings in the Wild

Mauro Pelucchi
Vice President Data Science @ Lightcast

- About 45% of 2026 US job postings read as LLM-drafted. Turning point: AI inside recruiting software, not launch of ChatGPT. Europe trails US by 1-2 years.
- Adoption is an organisational decision. Many employers have almost none or most postings AI-drafted: a template or tool on for all.
- Can't observe the label? Create it. An LLM rewrote pre-AI postings, same facts and length, so pairs differ only in author: reliable training data. The LLM is the instrument, not the product.
- Detecting isn't measuring. Reliable numbers needs checks for shortcuts the model may learn (formatting, near-copies), a pre-ChatGPT control for real false positives, and prevalence correction. The result describes millions of postings, never a verdict on a single posting or recruiter.

The Journey to Enabling AI for Analytics at Bolt: "One Middle Layer, Every Agent"

Kevin Sutanto
Analytics Engineering Manager @ Bolt

- Enabling conversational analytics requires AI-ready data: clear data model design, good business process coverage, and a semantic layer acting as the source of truth for metrics.
- Standardized context is valuable. However, for tools targeting business users, a smooth experience, such as one-click installation, is a must.
- Evaluation and benchmarking are often overlooked, yet they are a crucial part of context engineering. Benchmarking should be part of the context development loop.

Data & ML Stage

digit
Powered by sTARTUp Day

Unexpected side effects of becoming a manager

Tuule Sõber
Engineering Manager @ Zendesk

- The biggest side effect of becoming a manager has been that I can now bench press three times as much weight.
- Working out, staying curious, and the ability to remain neutral have helped me become a better manager.
- Communication can be made easier through simple practices, such as putting the most important information at the beginning of a message, keeping track of your achievements, and regularly sharing them with your manager.

Foreign Object Detection in Estonian Mines: Rethinking the Problem from Classification to Detection

Yaroslava Mykhailenko
Data Scientist @ Enefit

- Great metrics can hide a model that learned the wrong thing. Our first wood classifier reached ~99% precision and recall, but Grad-CAM showed it wasn't looking at the wood at all. It relied on belt edges and background differences, a classic case of shortcut learning. Always check why your model is right, not just how often.
- Plan for the gap between lab and field. It's not a bug, it's the job. A model that passes validation has only passed the first exam. Ours did, and still raised 896 false alarms in its first week in the mine. Expect new edge cases in production, budget time for them, and design the rollout so you can learn from every false alarm.
- The model is the smallest part of the solution. What turned hundreds of false alarms a week into just three wasn't a better algorithm. It was the decisions around the model: checking several frames in a row before raising an alert, choosing the confidence threshold deliberately, and retraining on real data from the mine. In industrial AI, reliability lives in the system, not in the model alone.

Throwaway code, lasting architecture

Klaus-Eduard Runnel
Software Architect @ Cybernetica

- Code and experiments are now cheap, so we can learn more about a domain, and far earlier, than before. Don't be afraid to throw away early code, but keep what you learned from it.
- Architecture is not cheap to change, because it follows the organization and its understanding of the domain. It is worth preserving, but should not be frozen. Agents don't change Conway's law; they make it visible sooner.
- A rewrite must preserve the intent behind decisions and the understanding of the domain. Both are hard to keep in agentic workflows.
- Keep people hands-on in the part of the system that gives it its value, or they lose the ability to judge what agents build.

Beyond the model: Operating verification ML at Wise

Iya Rõõs
Data Scientist @ Wise

- A strong ML model is only one part of a reliable production system. Evaluation, thresholds, monitoring, and ownership matter just as much.
- Testing one meaningful change at a time makes experiments traceable, releases safer, and failures easier to diagnose.
- The best candidate is not simply the one with the highest score. It must improve useful signals while protecting rare but important cases.
- Offline performance is only the first gate. Shadow testing, continuous monitoring, and recalibration help models remain reliable in production.

digit
Powered by sTARTUp Day

Building Agentic Systems for High-Stakes Decisions: Error Handling, Evals, Human Control and Trust

Vitalii Lakusta
VP of Engineering @ Pactum AI

- Deploying an agentic system to production for high-stakes business decisions requires planning how customers will trust it and judge its accuracy, so they can make decisions while clearly understanding and assessing the risks.
- Don't put an agent into critical business processes on day one. Build trust step by step, for example by running the agent in shadow and advisory modes first so the customer can see how it performs before the agent directly affects the business process.
- Give customers control and a way to grade the agent's performance. This helps them understand its risks and benefits.
- Moving the agent into production is the customer's decision, made once the risks are assessed and its performance in shadow and advisory modes is well understood.
- We walked through an end-to-end example of how Pactum deployed its Alignment Agent for Fortune 500 customers, such as AstraZeneca. Alignment Agent is built into key procurement processes, where it delivers significant business value and has earned customers' trust in Pactum.

Causal Agentic Data Science

Carlos Trujillo
Principal Data Scientist @ PyMC Labs

- AI has achieved an unprecedented level of intelligence, making it possible for everyone to develop what they imagine.
- This makes software development fundamentals more important than ever. Well-designed systems that allow fast and reliable iterations can push the boundaries of what is possible. We don't need more intelligent models; we need better systems.
- With the right system, anyone can make breakthrough scientific advances or software developments, just as OpenAI or Anthropic have shown many times this year.

Who Decides When AI Decides? Rethinking Human–AI Delegation Beyond Automation

Anastasija Nikiforova
Associate Professor of Applied AI and Information Systems @ University of Tartu

- AI adoption is increasingly about delegation, not just automation. As AI — particularly more agentic forms of AI — moves from assisting humans toward recommending, coordinating and acting, the important question is not only what AI can do, but what organizations choose to delegate to it.
- Human–AI delegation takes different forms, from translator and explainer to resolver and integrator. These roles represent different ways in which AI can participate in organizational work — and each creates different requirements for oversight, risk management and governance.
- Capability does not equal authority. An AI system may be technically capable of performing a task without being given the authority to decide or act. Delegation is therefore an explicit governance decision: what is delegated, with what scope and authority, under what conditions, and with what possibility for human intervention or reversal.
- Responsible delegation requires more than keeping a human "in the loop." Meaningful human control requires appropriate appraisal before delegation, human validation scaled to impact, mechanisms to reclaim authority when AI is uncertain, auditable outputs, and explicit human accountability.



digit
Powered by sTARTUp Day

LLMOverflow Stage – Practical discussions and hands-on demos on building and working with LLMs and agents

The Hype-Reality Gap: Notes from the Daily Grind

Moderator: Ain Arend I Co-Founder & CTO @ 99math
Panelists: Hendry Sadrak I Co-Founder & CTO @ Modash, André Simões I Software Architect @ TEKEVER, Juri Gavshin I Principal Software Engineer @ Bolt

- AI is transforming software development, not replacing engineers. Coding is becoming faster, while human expertise, critical thinking, creativity, and decision-making remain essential.
- AI creates opportunities for people to grow beyond their existing roles. Employees can take on new responsibilities, develop skills faster, and focus on higher-value work. André Simões, Software Architect at TEKEVER, said: “Thanks to our AI frameworks, our analyst is now a product developer.”
- AI adoption requires trust, responsibility, and adaptability. Companies must balance increasing AI autonomy with human oversight, reliable safeguards, and continuous learning as technology evolves.

AI in the Wild: What It Actually Costs to Build with It

Ivan Elorza Tahum I Co-founder @ Amiqa Tech AB

- Measure the cost per finished task, not the price per token.
- AI models have no memory: every step re-reads the whole conversation, so long sessions get expensive.
- Agent orchestration (one planner, cheaper specialist agents) cut our AI bill by about a third.
- Put a spending cap on every API key.

LLMOverflow Stage

digit
Powered by sTARTUp Day

Anatomy of a Production AI Agent

Natia Kukhilava I Co-Founder & CTO @ Helio AI

- "Finished" doesn't mean "helped". Evaluating an agent by whether it completed a task, counted tool calls, or matched a regex isn't enough. You have to measure whether it actually helped the user.
- Multi-agent systems are often a trap. A single well-designed agent with a strong system prompt and good model abstraction usually beats a complex multi-agent setup.
- Best practices don't exist yet. The field moves too fast, so the real lessons come from shipping to production and learning from what breaks.

Beyond the Horizon: What does the future hold? Can we predict it?

Moderator: Mart Parve I Head of Software Engineering @ Hevi Optronics
Panelists: Agur Jõgi I Tech Enthusiast, former Pipedrive CTO, Ivo Kund I Co-Founder & CTO @ Closaria AI, Kristiina Kapten I Solution Architect @ SEB

- AI adds a new layer to programming, building on existing technologies and opening software development to more people.
- Productivity gains require redesigning workflows. Adding AI to existing processes is like replacing a factory's motor while keeping the old belts.
- Engineering fundamentals remain essential, while coding practices will evolve as agents write and read more code.
- Management requires closer attention to people and results. Leaders need to distinguish real contributions from apparent productivity and rethink how work gets done.

How AI is Transforming Software Development in Biotech

Immanuel Sanka I Bioinformatics Platform Engineer @ Icosagen Cell Factory

- With AI, everyone can work across the whole cycle, from formulating questions to deployment. Although AI supports research, methods, and building, framing the questions and meeting human needs still require human intervention.

In biotech, "sequence" is the product and it never leaves the building. Agents are becoming smarter, but thorough study and testing are still required for better understanding.



wise

MAKE A WISE MOVE



wise.jobs wisecareers wise



digit
Powered by sTARTUp Day

Workshops

What We Can Learn from Exemplary Python Documentation

Christian Heitzmann
Owner @ SimplexaCode

- Good documentation starts with knowing your audience. Decide who you are writing for, what level of detail they need, and who is responsible for maintaining the documentation.
- Sphinx makes Python documentation easier to create and maintain. Generate API documentation from docstrings, use NumPy or Google-style docstrings to keep things simple, and take advantage of Sphinx's many extensions.
- Learn from existing documentation instead of starting from scratch. NumPy, pandas, and Matplotlib show how to use structured layouts, code examples, admonitions, mathematical formulas, and plots. Looking at their documentation source code reveals how these features are implemented.
- Choose your documentation tools based on your project's needs. reStructuredText and Sphinx are a natural choice for Python, while AsciiDoc is worth considering for projects involving several programming languages. Both approaches have advantages and limitations.

Building Kubernetes-Native Python Applications

Lauri Võsandi I Founder @ Codemowers,
Erki Aas I Senior Product Engineer @ Baltic LEI

- Software development is now declarative
- Prompting is dead, control loop engineering is next https://github.com/codemowers/agent

Getting Started with MCP, ADK and A2A

Karel Jakubec
Customer Engineer @ Google

- Participants learned terms such as skill, MCP server, agent, and A2A (agent-to-agent), and created a simple multi-agent system in Google Cloud.

Workshops

Own Your LLM: Lessons from Building a Local AI Agent

Heili Aavola
Junior Research Fellow in Machine Learning @ Institute of Computer Science, University of Tartu

- An LLM alone is not an AI agent. An agent harness, such as Hermes Agent (https://hermes-agent.nousresearch.com/), extends a locally hosted model with tools, memory, and multi-step task execution.
- Match the model to your hardware. Consider three factors: (1) memory capacity, (2) inference speed, and (3) model architecture. Memory determines what fits, while compute and memory bandwidth affect prompt processing (prefill) and token generation (decode), respectively. Dense and mixture-of-experts (MoE) models have different memory and computational requirements.
- Build systems in which the model is replaceable. New open-weight models appear frequently. Keep the model separate from the agent harness, user interface, and application logic.
- Test models on your own tasks. Leaderboards such as Artificial Analysis (https://artificialanalysis.ai/models?model-filters=open-source) provide an indication of general capabilities, but cannot determine which model is best for your application. Evaluate the actual quantised model on your hardware using representative tasks.

Beyond the Demo: Taking AI Agents to Production—Securely

Alexandru Neagu
Cloud Engineering Director @ Oracle

- A convincing demo is only the beginning. Moving an AI agent into production raises engineering questions about how it connects to enterprise data and operates safely.
- Build with production requirements in mind. The workshop uses open-source components, MCP, and an OpenAI-compatible API to create a small agent and examine the controls around it.
- Make the work reusable. Practical patterns, a reference architecture, a production checklist, and cloud account access support continued experimentation after the workshop.

Train your design eye

Elis Laasik
UX/UI Designer and Developer @ Codeborne

- Alignment. Everything in your UI needs to be aligned with something.
- Size. Define a hierarchy. Use size and boldness to highlight more important elements.
- Spacing. Items closer to each other seem more related and vice versa. Use a hierarchy.
- Color. Less is more. Use a monochrome color palette if you don't know what to do with color.

You Are the Harness. Until You Build One

Kirill Linnik
Principal Software Engineering Manager @ Microsoft

- If you are copying one agent's output into another, you are the bottleneck of your own AI setup. Fix the harness around the model, not the model.
- Build your own agent harness only when you can name exactly what the ready-made ones don't give you — for us it was determinism: small, focused agents, hand-offs decided by code, a human at the gate.
- Keep the "agentic exhaust" — every thought, tool call and decision your agents produce. It is your audit trail, and it is what makes agents learn from one run to the next.
- Don't start by building; start by trying — and when you do build, share the spec, not just the code.

Practical examples from an Agentic First company since 2023 with Soldera

Stenver Jerkku
CEO & Co-Founder @ Soldera

- AI should be able to do everything, but ACL is still key.
- For everyone who uses AI agents, give them only the minimum access they need.
- Leverage AI to accelerate business growth and stay competitive, while maintaining sound processes and following best security practices.
- Have automated gates and safeguards built into process and fail quickly when needed.



Thank you!

The biggest software conference in Estonia, Digit, brought together curious minds, inspiring speakers, and passionate tech professionals in Tartu.

A huge thank you to everyone who made Digit 2026 possible, including our speakers, participants, partners, volunteers, and the entire organising team!

We hope to see you again at **Digit 2027** on 29 October 2027! Before that, join us at **sTARTUp Day 2027** on 17–19 August 2027.

digit
Powered by sTARTUp Day

digit
Powered by sTARTUp Day

PARTNERS

Bolt · WISE · TARTU · ORACLE · playtech people · pipedrive · SEB · algorütm · Microsoft · // kood · TEKEVER · PyData · modash · CYBERNETICA · Google Cloud

UNIVERSITY OF TARTU
Institute of Computer Science

SUPPORTERS

Balbiino · DORPAT · A.LeCoq · For Me · SHROOMWELL · iDeal by C&C · kadarbiku · HOTELL SOHO · LUX EXPRESS · kohe · Maris Gilden · VÄIKE VÄÄNIK · TERE! · deary · mobilab. · intera · framery · Fresmy

Proudly presenting the community behind Digit 2026. Thank you!