

Join Leyoda — Where Your Code Actually Matters

COMPANY DESCRIPTION

Leyoda is Europe's first AI-powered investor-startup matching platform — a discovery engine where investors swipe through startup profiles, our AI surfaces the best matches, and real deals happen. The platform is live, processing real startup data and investor interactions across Europe. We're now looking for interns and early hires who want to ship real features to real users from day one, across a modern full-stack environment (Java, React, Python AI, Docker).

APPLICANT PROFILE

We're looking for curious, self-driven builders — students or recent graduates in CS, Data Science, AI, or related fields who have built something outside of coursework. You should be a power user of AI tools — we develop with AI agents that autonomously run validation pipelines, enforce quality gates, and orchestrate multi-step workflows. You don't need to know our exact stack, but you need to learn fast, stay calm when things break, and care about quality. We value initiative over experience and resourcefulness over credentials.

THE STORY BEHIND THE STACK

This isn't a class project. It's a live product with real users, real data, and real infrastructure. **Alexander** and **Sebastian** came up with the original concept and recruited **Alexandru** as the technical cofounder to build it. The entire platform — backend, frontend, infrastructure, CI/CD, security, real-time systems — was architected and shipped by Alexandru. **Roy**, another cofounder, brought the Signal Engine concept from the environment we operate in. You'll be working directly with that level of ownership and execution.

Now we're looking for people who want to help take it to the next level.

The stack, if you're curious:

- A Java backend with **22 REST controllers**, 25 business services, and 48 database migrations
- A React frontend with **47 pages**, 74 components, 51 API routes, physics-based swipe animations, and a custom design system
- A Python AI Signal Engine with **62 modules**, a 7-stage pipeline, and 4 web crawlers that process thousands of research papers and scores investment opportunities
- Zero-downtime blue-green deployments, a **9-job CI/CD pipeline**, 6 Docker services, PostgreSQL, NGINX

You don't need to know all of this. You just need to be able to learn it.

OPEN POSITIONS

01 Full-Stack Product Engineer

What you'd do: Build features that go all the way from the database to the user's screen. You'd write backend logic in Java, create frontend interfaces in React, set up database tables, write tests, and push it live — all in the same day. No waiting around for someone else to finish their part. You own the whole thing.

A TYPICAL WEEK MIGHT LOOK LIKE

- **Monday:** Design a new filtering feature for investor search
- **Tuesday:** Write the database migration and backend API endpoint
- **Wednesday:** Build the React UI with form validation and loading states
- **Thursday:** Write tests, fix edge cases, get it through the CI pipeline
- **Friday:** It's live. Real investors are using what you built this week.

YOU'D BE GREAT AT THIS IF

- You've built something that runs — a side project, a hackathon entry, a tool you actually use
- You enjoy figuring out how things connect, not just working on isolated pieces
- The idea of shipping a feature end-to-end excites you more than specializing in one tiny corner
- You learn fast and don't mind picking up a new language or framework when needed

Java, Spring Boot, PostgreSQL, Next.js, React, TypeScript, Tailwind CSS, Docker

02 AI / Data Engineer

What you'd do: Make Leyoda's brain smarter. We have an AI pipeline that reads research papers, extracts investment signals, groups them into themes, and scores opportunities on a 7-factor rubric. Your job is to make every step of that pipeline better — sharper signal extraction, smarter clustering, more accurate scoring.

A TYPICAL WEEK MIGHT LOOK LIKE

- **Monday:** Analyze why certain signals are being missed in the extraction step
- **Tuesday:** Rewrite the LLM prompt and test it against 50 sample papers
- **Wednesday:** Experiment with a different embedding model for better clustering
- **Thursday:** Build a new web scraper to pull patent data as an additional signal source
- **Friday:** Compare the old and new pipeline outputs — measure, don't guess

YOU'D BE GREAT AT THIS IF

- You've worked with data in any form — Kaggle, a university ML project, a personal data analysis, anything
- You understand that most of AI work is cleaning data and designing good systems, not tweaking hyperparameters
- You're comfortable with Python and curious about NLP, embeddings, or LLMs
- You like measuring things. "It seems better" isn't good enough for you — you want numbers.

Python, NLP libraries, sentence-transformers, scikit-learn, LLM APIs, SQLite, FastAPI

03 Frontend Engineer — Interaction & Performance

What you'd do: Build interfaces that feel premium. Leyoda's swipe cards run at 120 frames per second with physics-based animations, glass-morphism effects, and 3D perspective transforms. You'd own this experience — making it smooth on phones, tablets, and desktops, and building new interactive components that make investors want to keep swiping.

A TYPICAL WEEK MIGHT LOOK LIKE

- **Monday:** Profile a performance bottleneck in the swipe card animation on mobile
- **Tuesday:** Rewrite the animation to use requestAnimationFrame instead of CSS transitions — 2x smoother
- **Wednesday:** Build a new multi-step onboarding form with live preview of the startup card
- **Thursday:** Make it responsive — different layouts at mobile, tablet, and desktop breakpoints
- **Friday:** Polish loading states, empty states, and error states until everything feels intentional

YOU'D BE GREAT AT THIS IF

- You notice when an animation isn't smooth. It bugs you. You want to fix it.
- You care about how things feel, not just how they look
- CSS isn't something you fight with — you actually enjoy it (or at least respect it)
- You've used your own projects on a phone and fixed what felt wrong
- You have taste. You've rebuilt something not because it was broken, but because it wasn't good enough.

Next.js, React, TypeScript, Tailwind CSS, CSS animations, Framer Motion, Web Performance APIs

04 DevOps & Infrastructure Engineer

What you'd do: Take a system running on a single node with blue-green zero-downtime deployments and evolve it into something that can handle 100,000 users without breaking. Add observability so we know what's happening. Add reliability so nothing fails at 3am. Build the foundation that lets everything else scale.

A TYPICAL WEEK MIGHT LOOK LIKE

- **Monday:** Set up structured logging so we can actually trace a request from frontend to database
- **Tuesday:** Build a Grafana dashboard that shows API response times, error rates, and database query performance
- **Wednesday:** Add automated rollback to the deployment pipeline — if health checks fail, go back instantly
- **Thursday:** Tune PostgreSQL connection pooling and analyze slow queries
- **Friday:** Write a runbook for the most common failure scenarios

YOU'D BE GREAT AT THIS IF

- You've SSH'd into a server and fixed something. Even once. Even if it was your Raspberry Pi.
- You think about what happens when things go wrong, not just when they go right
- You like automation. Doing something manually twice feels like a waste.
- Dashboards and metrics make you happy. Servers with no monitoring make you nervous.
- You've used Docker and want to go deeper — networking, health checks, multi-service orchestration

Docker, Docker Compose, NGINX, GitHub Actions, PostgreSQL, Linux, Bash, Prometheus/Grafana

Why This Is Different

Your code goes to production.

Not to a sandbox. Not to a demo. To a live platform used by real investors across Europe. We deploy multiple times per week. What you build on Monday can be serving requests by Tuesday.

You'll see the whole picture.

Most internships put you in a corner of a giant codebase you'll never understand. Here, you'll touch the database, the backend, the frontend, the AI pipeline, and the deployment infrastructure.

The problems are real.

How do you make a swipe animation feel smooth on a 5-year-old Android phone? How do you extract reliable signals from a messy research paper? These aren't textbook exercises.

You'll develop with AI agents.

We orchestrate AI agents that bootstrap full project context, run autonomous validation pipelines, and enforce enterprise-grade quality gates — all without manual intervention.

You'll work directly with the cofounders. No layers of management. No Jira tickets from someone who's never read the code. Direct collaboration with the people who made every architectural decision — because they can explain every one of them.

What We Actually Care About

We don't care about your GPA. We care about one thing:

Have you built something?

A Discord bot. A mobile app. A web scraper. A personal website that's actually deployed. A script that automates something annoying. It doesn't need to be impressive — it just needs to exist. **This environment isn't for everyone.** Things move fast, expectations are high, and you'll be shipping real code from day one. But if that excites you more than it scares you — keep reading.

THE MINDSET THAT WORKS HERE

Get curious, not scared, when something breaks. A stack trace is a treasure map, not a stop sign.

Try things before asking. Google it, read the docs, experiment, fail, try again. Then ask. That sequence matters.

Care about quality without being told to. You fix the thing that looks slightly off, not because someone filed a bug, but because you noticed.

Don't box yourself in. "I'm a frontend person" or "I only do Python" — that thinking doesn't work here. You're an engineer. You figure things out.

Use AI like a power tool, not a crutch. You know how to prompt, orchestrate, and verify AI output — not just copy-paste from ChatGPT and hope it works.

Want to do something that matters. You're here because building a real product sounds more interesting than another course project.

We're only hiring a small number of people. If any of this sounds like you, come talk to us at the fair. Or send an email via contact@leyoda.eu — no formal cover letter needed. Tell us what you've built and what you'd want to work on.

We'd rather see your GitHub than your CV.

Leyoda — where builders become operators