CV

# ALEKSEI RYBKIN
## AI ENGINEER

I build language models into working products and take them to production. In a week built an LLM contract review service with verified quotes and a Word and PDF protocol; designed a headless content pipeline on the Anthropic API with an MCP connector — 500+ posts from one person; moved an online store onto Next.js with a 1C exchange, a Python parser and monitoring. I design the architecture myself, work with agents through Claude Code and accept results only after automated tests and code review.

t.me/oXaoc  neurodivergent@yandex.com  github.com/chaoseye
linkedin.com/in/rylex  rybkin.dev
Case studies: rybkin.dev/cases/contract-review, rybkin.dev/cases/roton · How this site is built: rybkin.dev/how

### EXPERIENCE

OCT 2026
PROJECT-BASED
### AI Engineer

**Contract review — AI service for a B2B supplier** github.com/chaoseye/contract-review

- Built a web service that reviews a buyer's supply contract with an LLM and issues a protocol of disagreements in Word and PDF: from the brief to a working system on the server in a week, working with Claude Code.
- Made the model's answer verifiable: a strict JSON schema (structured outputs) with one re-ask, every quote matched against the contract text — exactly, then fuzzily — and clause numbers and annex references checked against the document markup.
- Connected models through an OpenAI-compatible API (RouterAI with Russian-only servers, OpenRouter, Ollama, vLLM), the Anthropic API with streaming and prompt caching, and Codex CLI; the admin keeps prompt versions with a diff, rollback and a trial run on a contract.
- Read real documents — docx from XML with Word's numbering restored, doc and rtf through LibreOffice, PDF through pdfplumber, scans through Tesseract cell by cell — and built the protocol as a business document. A contract file is deleted once read and the review after 24 hours, texts never reach logs or backups; 450+ automated tests.

JUN – AUG 2026
PROJECT-BASED
### AI Engineer

**Roton — medical supplies store** roton-med.ru

- Moved a medical supplies shop off a hacked Bitrix site into a clean build on Next.js and WordPress/WooCommerce. Page load time (LCP) fell from 4.1 to 1.1 seconds.
- Set up the automatic 1C exchange and wrote a Python parser for product attributes: around 1,100 products with stock and photos now update on the site by themselves. By the client's estimate that freed 20 hours of a manager's time a month.
- Kept the search traffic through the change of engine: every old address leads to the right page, and search positions were back where they had been within three weeks.
- Built search with filters and ordering through a chat on the site: the message goes to the managers in Telegram, the reply comes back to the visitor in the same chat.
- Set up monitoring of the exchange and backups with restore checks. Prepared the project for handover to another developer: documentation, automated tests, one-command deployment.

Built in tandem with Claude Code: I designed the architecture myself, broke the work into tasks for the agent and accepted results only after automated tests and code review. That is what let one person carry the whole migration.

JAN — AUG 2026

## Product Manager

### OneTwo — Telegram games `onetwo.gg`

- Designed and shipped a headless content generation pipeline. Claude writes the texts through the Anthropic API; NanoBanana makes the images through Higgsfield's remote MCP server. The server is attached to the same call through an MCP connector: the model itself triggers generation and polls the asynchronous job until it is ready. A prompt template per format, few-shot built from the brand's own reference texts. A person selects and finishes.
- Five months produced 500+ posts, 100+ a month, from one person instead of three — a content manager, a designer and a marketer: about 1.5M roubles of payroll a year. 56% of drafts shipped unedited, and time per post fell from four hours to twenty minutes.
- Wrote the specifications for developers and designers: feature breakdowns, game mode logic, acceptance criteria.
- Found divergences between the specification and the implementation during testing — 30 bugs closed before release.
- Designed two battle passes and a reward ladder with parameters, dependencies and a target retention metric. D7 retention went from 22% to 31%.
- Turned the results of community surveys into roadmap priorities.
- Ran a competitive analysis of the Telegram games market — the findings fed features and promo campaigns.

JUN 2024 — DEC 2025

## Project Manager

### Madnomad Design — design and marketing agency `madnomad.design`

- Moved advertising graphics onto generative models: banners for a property developer are assembled in NanoBanana and Recraft to the brand's style. Four layouts a day against one, four times less time per layout. Adapting to platform formats and handling revisions stopped costing a separate round with a designer.
- Automated the routine in Jira and Confluence: issue templates, automation rules, guides. Around 40 manual steps a week left the process.
- Formalised requirement intake: checklists and briefs cut post-delivery rework by 60%.
- Assembled one documentation base in Confluence — onboarding went from ten days to three.
- Kept the roadmaps and coordinated five parallel projects.

JAN 2023 — MAY 2024

## Business Assistant

### EMCD Tech — IT and fintech, Saint-Petersburg `emcd.io`

- Automated the handling of calls: meeting transcription with an automatic summary and a list of decisions. Manual minutes used to cost 14 hours a week of the participants' time; now it is five minutes of proofreading after the call. Agreements stopped falling between meetings, and searching past discussions takes seconds instead of digging through chats.
- Simplified work with Google Sheets: data consolidated between sheets automatically, reporting forms filled in on their own. Twelve manual steps left the weekly cycle, along with the copying between documents and the errors it produced.
- Collected and visualised data, prepared analytical reports for management decisions.
- Initiated and pushed the Notcoin integration: gathered the metrics from past integrations and the case for it, and brought warm contacts. The integration shipped.
- Researched the competitive market and segmented the audience for the company's positioning.

## EDUCATION

2020 – 2024

BACHELOR'S DEGREE

### Software Engineer

**SPbPU — Peter the Great St. Petersburg Polytechnic University** `spbstu.ru`

Degree programme: Fundamental Informatics and Information Technologies.

## STACK

### LLMS AND AGENTS

Claude Code, NanoBanana, Anthropic API, OpenAI API, MCP, Prompt Engineering

### BACKEND AND DATA

Python (Advanced), FastAPI, Node.js, REST APIs, SQL (Advanced), PostgreSQL, MySQL, Redis, Meilisearch

### FRONTEND

TypeScript, React, Next.js

### INFRASTRUCTURE AND INTEGRATIONS

Docker, Linux, Cloudflare Workers, 1C (CommerceML), Telegram, n8n