

Maksym Holovanov — CV

版本 2026.2

科技领域 25 年以上 — 编程、Web 开发、网络安全是我工具箱的核心,加上一系列真正能推动事情前进的能力。驱动力:推动 AI 与技术进步,基于多年经验将自动化与生产力推向新的层次。

经历与技能

- **1992** → 我从 8-bit Dendy 主机开始接触科技,玩了 Contra、Mario 等经典游戏。
- **1996** → 在 ZX Spectrum 上用 BASIC 迈出了编程的第一步。一像素一像素,写出了我的第一款主机游戏。
- **1998** → 深入汇编语言,获得对各种算法的更深理解,夯实了我的技术基础。
- **2001** → 进入 Web 开发,涉足 PHP、HTML、CSS。21 世纪初的 PHP 与如今完全不同。这段时间也精进了 Photoshop 技能。
- **2002-2004** → 在 **STEP IT Academy**(乌克兰) 学习 Web 设计,这也是我第一次接触 C++ 和 C# 语言,后来发现线上有更好的学习资源。
- **2004** → 投入大量时间到金融服务、支付系统和安全分析的开发上。
- **2008** → 进入流量分析、Wireshark,以及第一次面对客户端-服务器 DDoS 攻击。由此走向 Security Audits、Linux 系统管理与网络安全。
- **2010** → 从传统主机转向 Linux 服务器,聚焦 Linux 系统管理、Bash、VIM 等技能。
- **2011** → Python 成为我道路上的重要部分 — 不禁自问为何在 PHP 上耗费那么多时间。同时也开始用 GIT 进行版本管理。
- **2013** → 接受了 Django 框架,并对 Docker 有了认知。
- **2014** → 加密货币与区块链时代开始,聚焦于金融项目的 API 与 REST 后端开发。
- **2015** → Vue.js 改变了我的前端开发体验,提升了 JavaScript 能力。之后扩展到 TypeScript。
- **2016** → 进入 Machine Learning (ML) 领域,搭建了第一批模型,并探索 Jupyter Notebook。大量阅读,完成了 AI、ML、DL 与 XGBoost 的课程。
- **2017** → 从这一年起,我开始以 DevOps Engineer 身份工作,并在这一角色中积累了大量经验。
- **2020** → Kubernetes (K8s) 与 Helm 终于进入我的世界。
- **2021** → 开始在测试阶段试用 chatGPT,花时间理解其 API。
- **2022** → 在 Infrastructure as Code 与日志、数据分析方面扩展技能 — 掌握了 Terraform 与 ELK Stack。
- **2023** → 主编辑器从 Vim 切换到 Emacs,服务器上的快速编辑仍然用 Vim。也开始每天使用 Claude 来增强研究、分析与整体生产力。当然还有 MCP。
- **2024** → 把键盘布局从 QWERTY 切换到 Dvorak,使用无字母键的 Kinesis Advantage360 Pro,反而帮助我更快掌握。Kinesis 是我最喜欢的品牌,会继续使用。
- **2025** → 开始学习 Nix,转向声明式配置环境。逐步把 Homebrew 与 APT 等命令式工具替换为完全由 Nix 管理。Claude、Gemini、OpenAI、Deepseek 的 AI 模型游戏也开始了 — 我觉得非常有趣。
- **2026** → 我搭建并发布了 **TRIADA** — 一套从 **Agile Results (AR)**、**GTD** 与 **The 12 Week Year** 综合而来的年度规划系统。每天、每周、每月、每季度三个成果。每个尺度,同一个形状。

资历摘要

- **编程语言:** Python、JavaScript、TypeScript、HTML、CSS、SASS、**Nix**。过去: C#/C++、Delphi、BASIC、Assembly、PHP。

- **Web 开发:** 精通 Django、Vue.js、Vite;使用 REST API、GraphQL、HTML、CSS 与 SASS 构建稳健的 Web 应用。
- **数据库管理:** 熟练使用 Postgres、MySQL;在 Python 中使用 Pandas 与 NumPy 进行数据处理。
- **DevOps 与容器:** Docker、Podman、Kubernetes (k8s)、Helm、Git、Mercurial、Terraform、CI/CD、ELK Stack、Sentry、**Nix**。
- **系统管理:** Linux、Bash、ZSH、Fish Shell、NGINX、Linux 管理、Raspberry Pi。
- **AI 与数据科学:** AI、Machine Learning (ML)、Deep Learning (DL)、XGBoost、Jupyter Notebooks。每日使用 **Claude**、**Gemini**、**Codex** 与 **Warp**。
- **测试与开发工具:** Poetry、PIP、PyTest、Scrapyd,以及聊天机器人与 Web 爬虫。
- **安全与审计:** Security Audits、网络安全审计、系统完整性、客户端-服务器 DDoS 攻击处理。
- **其他:** Dvorak 键盘布局用户。

教育

我的基础来自 **STEP IT Academy(2002-2004,乌克兰)**,在那里钻研 Web 设计,并接触 C++ 与 C#。后来真正的学习空间是线上 — 我读了大量专业书籍,完成了 [Udemy](#) 与 [Coursera](#) 上的多门课程。

但我的主要专业能力,来自一路上的实战项目。

工作经历

在我的 IT 职业生涯中,从来没有「只做一个项目」或「每天同样套路」的时期。单调与我的风格格格不入 — 感觉像是进展的停顿。同时做多个项目是常态,典型的自由职业风格。

在后端开发与系统工程领域,我专注于以稳定性为核心的可靠系统。重点放在稳健的架构上 — 它是金融 API、支付系统与平台的脊柱。除此之外,我也负责服务器管理与 DevOps。与 Web 设计不同,这类工作很少可见 — 它讲的是长期稳定。

我的大部分时间花在高保密性的项目上,这是金融科技的常态。因此无法分享具体细节,我的后端项目也不会出现在这份 CV 中。也许将来我会回到 Web 设计,展示一些可见的作品 — 横幅、Logo。但这不是我的主轴。

方法论

在 IT 之路上,我接纳了塑造工作、提升效率、推动协作的核心原则与系统 — 最终带来顺畅、成功的项目执行。

- **Getting Things Done (GTD):** 我整理任务与项目的首选系统 — 确保不漏一件事,并保持头脑清晰,留出创造性工作的空间。
- **Agile Results (J.D. Meier):** 通向个人与职业卓越的指南 — 高效管理时间与精力,明确优先级,达成平衡的结果。
- **PARA 方法:** 优化数字信息组织 — 通过整理存储与访问方式最大化生产力。
- **敏捷方法论:** 推动协作、适应性与有价值软件的交付 — 聚焦客户满意度,实现持续改进。
- **看板:** 可视化工作、限制在制品、有效管理工作流 — 让项目保持在轨道上的关键工具。
- **12 Week Year (Brian Moran):** 加速目标达成 — 借助锐利的聚焦,在 12 周内完成别人一年才能完成的事。

沟通语言

- 英语、乌克兰语 (母语)、俄语。

- A2 德语 (学习中)

联系方式

首次联系请通过 **Threema**, ID **X88K34NK**, 或通过 maxclax.com 发送消息。

X (Twitter): <https://twitter.com/maxclaxOS> **LinkedIn:** <https://www.linkedin.com/in/maxclax>

其他

更多信息请访问: <https://maxclax.com>