

KRISTINA KOLESNYK

FRONTEND DEVELOPER

Frontend Developer focused on making technology simple and enjoyable for people. I build clean, responsive web experiences with React and JavaScript and ship end-to-end features with Node and PostgreSQL. Committed to performance, accessibility, and clear communication — turning complex requirements into simple, maintainable solutions. I take ownership, collaborate closely with design and product, and keep users at the center. For me, every project is an opportunity to grow and bring value to users.

TECHNICAL SKILLS

Frontend: React.js, TypeScript/JavaScript (ES6+), HTML5/CSS3, Tailwind CSS, Sass/SCSS

Backend & APIs: Node.js, Express.js, RESTful APIs, WebSocket (real-time)

Databases & ORM: PostgreSQL, MySQL, Sequelize, SQL

Tools, VCS & CI/CD: Git, GitHub; IntelliJ IDEA, WebStorm, Visual Studio, VS Code; CI/CD pipelines (GitHub Actions)

Environments: Linux (Ubuntu), macOS, Windows

UX & Practices: UX/UI principles (usability, IA), Accessibility (WCAG basics), Performance optimization, Code review, Git Flow

Additional: Java (fundamentals/OOP, Collections)

PROFESSIONAL EXPERIENCE

AI KNOWLEDGE ENGINEER | Hola (2024 – 2025)

hola.org

- Built the knowledge base for an LLM support agent: curated Toolip, Oculus Proxies, and Browser.ai docs into a Git-versioned, structured corpus with standardized metadata (YAML front matter), stable IDs, and internal cross-links. This enabled reliable retrieval and easier diffs/reviews across releases.
- Prompting & guardrails: built intents, prompts, and fallback flows; created evaluation set (FAQ and edge-case questions with expected answers); reviewed model answers and iterated until they were accurate, consistent, and on-brand. Raised answer accuracy from ~62% to ~78% (+16 pp) over 4 sprints.
- Deflection & response time: routed recurring questions to the AI layer based on up-to-date docs and intents, deflecting ~28–35% of high-volume FAQs and reduced first-response time from hours to seconds for the topics it covers.
- Developer-grade docs: produced product overviews, conceptual guides (e.g., SOCKS5 Proxies; Shared vs Dedicated; Global vs Search Engine), error troubleshooting, integration guides (proxy managers, scraping tools, automations), and billing info. Improved structure, naming, and examples increasing self-serve problem solving by ~22–27%.
- Web UX improvements: reviewed product websites, proposed UI/UX changes, and created actionable tickets in Monday; identified bugs and collaborated with devs on fixes and code contributions.
- Email development: converted Figma designs into production-ready emails for Oculus/Toolip; ensured cross-client rendering and followed deliverability best practices.

Tools/Stack: Git/GitHub, Help-center/CMS (e.g., Mintlify, Zendesk, Intercom), Figma-to-HTML, basic REST concepts, Monday, FAQ tools.

- Pre-release verification (web + devices): validated commits before production rollout across web, iOS, Android, and smart TV apps; executed release runbooks (standardized procedures) and CLI validation checks (scripts/commands) on Ubuntu/macOS; blocked risky releases and coordinated fast fixes.
- Automation: built lightweight scripts (Bash/Node.js) to automate sanity/regression steps, log capture, and API probes; reduced pre-release verification time by ~30–40% and improved consistency across environments.
- Monitoring & incident response: operated Grafana dashboards/alerts, correlated metrics, and escalated with context; first-response time (FRT) to engineering <3 min, MTTR ~1–2 min on P1/P2 incidents.
- Device & environment coverage: validated critical flows across real devices and target OS/browser matrices; caught environment-specific regressions early.
- Deployments & rollouts: collaborated with R&D and customer teams to ensure smooth releases and clear rollback criteria.
- Production stability: monitored and optimized production environments on Linux and Windows, ensuring reliable service for ~20k B2B clients.

Tools/Stack: Grafana, Git, Ubuntu/macOS terminal, real device testing (iOS/Android/TV), browser devtools, release runbooks, basic API checks.

PROJECTS

Ranker – Full-stack web app (similar in idea to **Tiermaker**) that helps users make decisions by ranking items in lists and head-to-head tournaments.

- Users create lists (e.g., movies, books, products), rank items by multiple criteria, and compare them in interactive tournament brackets
- Responsive React frontend with real-time score updates, animations, and visual feedback
- Secure backend with RESTful API, authentication, and PostgreSQL for storing lists, rankings, and tournament results.

EDUCATION & CERTIFICATIONS

BSc in Software Engineering | Afeka Tel Aviv Academic College of Engineering

Web Developer Certification (Full-Stack Web Development) | Zero To Mastery Academy