Vladislav Kharitonov

Full Stack Software Engineer | .NET | Azure | React / Angular | AI-Assisted Development

Website: vladkharitonov.com | LinkedIn: linkedin.com/in/vlad-kharitonov-aa2607198 | GitHub: github.com/vladkharitonov87

SUMMARY

Full stack .NET engineer with 19 years in software development: system architecture, backend development and API development on Azure (ASP.NET Core, Entity Framework Core, microservices with MassTransit and Azure Service Bus, Terraform), with React and Angular on the front end. Built and modernized enterprise web systems for clients from the USA, including performance optimization, legacy system modernization and production troubleshooting with Dynatrace, Opsgenie and Application Insights. Worked as a team lead - technical leadership, code review, mentoring and technical interviews. Introduced AI-assisted development end to end on the Relay platform (Claude Code, LLMs, AI agentic SDLC), where a task goes from backlog to production in 1-2 days, and ships own AI products: Voice Helper for Claude, hands-free voice control for the Claude desktop app, SQL Server gateway and a Google Meet autopilot. English B2.

PET PROJECTS

Voice Helper for Claude - voice-helper-for-claude.com (2026, Windows desktop). Hands-free voice control for the Claude desktop app: the wake name "Claude" / "Клод", dictation, auto-send and eight voice commands, recognised fully offline in English and Russian. Built the product alone - the app itself, offline command recognition, tray and settings, an installer with checksum-verified updates and the site. **Tools:** C#, .NET, WPF, Vosk offline speech recognition, Win32 API and UI automation, GitHub Actions, Windows installer

SqlGate - SQL Server gateway (2026, Windows). A gate between developer tools and SQL Server databases: parses every request with the SQL Server parser, checks it against per-environment allow / ask / block rules and runs it, asks the operator or refuses it; passwords are sealed with DPAPI and the channel is HTTPS with a pinned certificate. **Tools:** C#, .NET, Windows Service, WPF, SQL Server ScriptDom parser, HTTPS with certificate pinning, DPAPI, CLI client, installer with firewall rules

Meet Autopilot - Google Meet auto-opener (2026, Windows). Reads the Google calendars of one or more accounts and opens each meeting Meet link in the right account a few minutes before the start, in a background browser window with the microphone and camera off. **Tools:** C#, .NET, Google Calendar API and OAuth, Chrome DevTools Protocol, Windows notifications, DPAPI, tray application, installer with self-update

TECHNICAL EXPERTISE

Programming Languages: .NET/C#, TypeScript, JavaScript, SQL, C/C++, HTML, CSS, SCSS, XML, XSLT, SSRS

Operating Systems: Windows, Linux

Databases: MS SQL Server, PostgreSQL, MySQL, Cosmos DB, Redis

Backend: ASP.NET Core, ASP.NET MVC, ASP.NET Web Forms, Entity Framework (Core), MassTransit, .NET Aspire, Azure Functions, REST API, LINQ, AutoMapper, Autofac, Ninject, Emit Mapper, COM/DCOM, WIN32 API, Microservices

Frontend: Angular, NgRx, React & Redux, jQuery, AJAX, HTML/CSS/SCSS

Cloud & DevOps: Azure (Container Apps, Functions, Service Bus, Blob Storage, Key Vault, Application Insights), CI/CD, Terraform, Docker, Azure DevOps, GitHub Actions, Grafana

AI & AI-Assisted Development: Claude Code, LLMs, AI-Assisted Development, AI Agentic SDLC, prompt and context engineering, review of AI-generated code, MCP servers and agent workflows

Architecture & Engineering: System Architecture, Backend Development, API Development, Performance Optimization, Legacy System Modernization, Production Troubleshooting, Technical Leadership, Code Review, Mentoring, Technical Documentation

Testing: NUnit, Moq, Shouldly, Testcontainers

Tools & Practices: Git, Bitbucket, MS Team Foundation Server, SVN, JIRA, UML, Design Patterns, Code Review, Agile (sprint planning)

PROFESSIONAL EXPERIENCE

Mitrix Technology – Remote **May 2025 – Present**

Project: TicketVault Processor (Relay) (May 2025 – Present)

TicketVault Processor (Relay) is a platform for ticket brokers that automates the ticket lifecycle: it processes purchase confirmations from primary marketplaces, creates purchase orders in the broker's POS, synchronizes inventory, broadcasts listings to secondary exchanges and handles sales and delivery. The system is a set of .NET microservices on Azure that communicate via MassTransit and Azure Service Bus, with Angular front ends and Auth0-based authentication.

Senior .NET Developer (May 2025 – December 2025; June 2026 – Present)

- Introduced AI-assisted development across the whole task lifecycle - analysis, implementation, writing test cases and testing in the dev environment - automating about 90% of the work while keeping the key decision points under human control, which brought a task from backlog to production in 1-2 days
- Developing backend architecture and services
- Developing database scripts and migrations
- Integrating with third-party services
- Setting up Azure services and creating Terraform scripts
- Code review
- Writing unit/integration tests for core logic

Team Lead (January 2026 – June 2026)

- Leading the development team: sprint planning, task breakdown and estimation, distribution of work
- Reviewing code and pull requests, maintaining coding standards and technical direction across the microservices
- Acting as the main technical contact for the product owner, QA and the customer; clarifying requirements and refining the backlog
- Mentoring developers and helping new team members onboard
- Conducting technical interviews with candidates
- Staying hands-on: delivering features and fixing defects in the Web API, POS and Purchase services
- Established the two-week sprint process for a team of 11 people - planning, refinement, demos, retrospectives, one-on-ones and the rest of the ceremonies - and kept it running on schedule sprint after sprint

Tools, Environment, Languages and Technologies: C#, .NET, ASP.NET Core, Entity Framework Core, MS SQL Server, Redis, MassTransit, Azure Service Bus, RabbitMQ, .NET Aspire, Azure Functions, Azure Container Apps, Azure Blob Storage, Application Insights, Auth0, Angular, NgRx, REST API, OpenAPI/NSwag, Terraform, Docker, Azure DevOps, Git, NUnit, Moq, Shouldly, Testcontainers, AutoMapper

Coherent Solutions (formerly ISsoft) – Minsk, Belarus **August 2012 – May 2025**

Project: Group Manager (February 2025 – May 2025)

Software developer

Group Manager is a web application developed for the internal purposes of Coherent Solutions Inc. to automate the Group Management processes in the company. This application contains a set of business processes aimed at improving the performance of the company employees.

- Replaced manual HR workflows with one web app that runs the group-management cycle end to end, with no handovers between spreadsheets and emails

- Updated the database; developed backend and frontend services, the backend architecture and the API.
- Wrote unit tests for core logic; communicated with the customer and third-party partners.

Tools, Environment, Languages and Technologies: C#, Angular, SQL, ASP.NET Core, Entity Framework Core, HTML, SCSS, CSS, REST API, Moq, NUnit, Shouldly, Git, Bitbucket, JIRA, Grafana, Migrations, Visual Studio Code, MS SQL Server Management Studio

Project: Anytime Fitness – Corporate Wellness (September 2021 – February 2025)

Software developer

Corporate Wellness helps employers and partners engage and activate members to reward healthy behaviors to change lives for better health.

- Designed the backend and database architecture; rebuilt the schema, indexes and stored procedures, which shrank the database and brought the slowest reports from minutes to seconds
- Set up Azure services for the migration (Key Vault, Blob Storage, Function App) and created Terraform scripts to update or create them.
- Cut incident detection from hours to minutes by wiring Dynatrace, Opsgenie and Slack into one alert path, so alerts reach the on-call engineer without manual polling
- Built the architecture of the React application with a responsive UI and responsive email notifications with Outlook support; configured settings for lower and production deployment.
- Set up GitHub Actions workflows integrated with Slack and JIRA; created Git repositories.
- Defined requirements, documented new processes, reviewed code, wrote unit/integration tests and communicated with the customer and third-party partners.

Tools, Environment, Languages and Technologies: C#, React Redux, Azure, SQL, ASP.NET Core, ASP.NET MVC, Terraform, HTML, SCSS, CSS, Media CSS, REST API, Moq, NUnit, Shouldly, LaunchDarkly, XML, jQuery, Git, GitHub Actions, Redis, JIRA, Dynatrace, Opsgenie, TeamCity, Kibana, Okta, MS Visual Studio, Visual Studio Code, MS SQL Server Management Studio, Azure Data Studio, GitKraken, MiniProfiler

Project: Anytime Fitness – Healthy Contributions (April 2015 – September 2021)

Software developer

Healthy Contributions is the system that provides health and fitness clubs across the nation with reliable and easy-to-use management solutions. Funds can be allocated to members directly, to the appropriate fitness institution, or to the health plan sponsor.

- Designed the backend and database architecture; rebuilt the schema, indexes and stored procedures, which shrank the database and let the client keep years of history online instead of archiving it
- Defined requirements, reviewed code, wrote unit/integration tests and communicated with the customer and third-party partners.

Tools, Environment, Languages and Technologies: C#, SQL, ASP.NET MVC, Terraform, HTML, SCSS, CSS, Moq, NUnit, XML, jQuery, Git, GitHub Actions, Redis, JIRA, Dynatrace, TeamCity, Kibana, Okta, MS Visual Studio, MS SQL Server Management Studio, Azure Data Studio, GitKraken, MiniProfiler

Earlier projects

VIAM Maintenance & Support – First Advantage [Verifications, Inc.] (February 2014 – April 2015). Maintenance and support of a pre-employment screening system; rebuilt the paginated and mobile reports in SSRS, so a client report is prepared with one run instead of by hand. **Tools:** JIRA, MS Visual Studio 2015, Team Foundation Server, XSLT, SSRS, XML

Evolution1 Lighthouse (January 2013 – February 2014). Health care insurance system for the USA; designed the architecture, the data model and the SSRS reports for members and their health plans, and delivered them on schedule. **Tools:** JIRA, MS Visual Studio 2012, Team Foundation Server, C#, ASP.NET 4, CSS, HTML, XSLT, SSRS, XML

RMSys (September 2012 – January 2013). Human resource management system; developed the architecture and design, models, views and controllers, fixed bugs and automated the HR paperwork,

so the department no longer does manual data entry. **Tools:** MS Visual Studio 2010, Team Foundation Server, C#, JavaScript, ASP.NET MVC 3.0, LINQ, Entity Framework, CSS, HTML, AJAX, jQuery, Autofac, Emit Mapper

SelfCom – Minsk, Belarus **November 2012 – September 2013**

BeCamOnline (January 2013 – September 2013), Software developer / Architect. Internet portal for online bloggers with the ability to conduct online conferences; requirements specification, architecture of the core, data/business/view layer, bug fixing, deployment - the portal was built from scratch and supported live online conferences. **Tools:** JIRA, SVN, MS Visual Studio 2012, SQL Server Management Studio 2012, Bitbucket, MVC 4, C#, CSS, HTML, AJAX, LINQ, jQuery, SQL, XML, Autofac, Emit Mapper, Entity Framework

ToNeSi (November 2012 – April 2013), Software developer / Architect. Web site that ranks other sites with the ability to leave comments; requirements specification, architecture of the core, data/business/view layer, bug fixing, deployment - the second portal built from scratch, from requirements to deployment. **Tools:** JIRA, SVN, MS Visual Studio 2012, SQL Server Management Studio 2012, Bitbucket, MVC 4, C#, jQuery, CSS, HTML, AJAX, LINQ, SQL, XML, Autofac, Emit Mapper, Entity Framework

BelEMN – Minsk, Belarus **August 2007 – September 2012**

UniCon (August 2007 – September 2012), Software developer / Architect. Platform for configuring the company's devices (UI and a COM server); architecture and design of the main program and of the COM server, bug fixing - the manual routine of device configuration was replaced by a template-driven UI and a COM server. **Tools:** MS Visual Studio 2008, C++, C#, COM, SQL, UML, XML

BEMN OPC Server (December 2010 – June 2011), Software developer / Architect. Server that connects to devices and data sources and translates their data into the standard OPC format; architecture and design of the server, bug fixing - the device data sources were unified behind one server, so a new device type is added by describing its data format instead of building a new integration. **Tools:** MS Visual Studio 2008, C#, COM, SQL

EDUCATION

Specialist degree (MSc equivalent), 2004 - 2009
Faculty of Information Technologies and Robotics
Belarusian National Technical University

LANGUAGES

English – B2 (upper-intermediate)