

Serhii B.

Senior .NET Software Engineer

SUMMARY

- A software engineer with 10+ years of experience in the IT industry who builds background services and APIs that are reliable and robust. - Skilled in C#, .NET Core, Apache Kafka, PostgreSQL, Docker, and Kubernetes, and know how the Windows operating system works internally. - Has worked with high-load microservices for the last 3 years and has experience building backend orchestrators. - Has experience working in both startup and corporate environments. - Has expertise in designing features, working closely with solution architects, and leading teams to improve the maintainability, scalability, and reliability of projects. - Upper-Intermediate English.

TECHNICAL SKILLS

Main Technical Skills	C#
Programming Languages	C#, C++, Delphi, Java, PHP, TypeScript
.NET Platform	ASP.NET Core Framework, ASP.NET MVC, ASP.NET Web API, Hangfire, .NET, .NET Core, TPL DataFlow
JavaScript Frameworks	KnockoutJS
Data Analysis and Visualization Technologies	Kibana
Databases & Management Systems / ORM	Aerospike, AWS ElasticSearch, FireBird, Microsoft SQL Server, PostgreSQL, SQL
UI Frameworks, Libraries, and Browsers	HTML, XML
Cloud Platforms, Services & Computing	AWS
Amazon Web Services	AWS ElasticSearch
Azure Cloud Services	Azure Service Bus
Platforms	Aerospike
Web/App Servers, Middleware	Apache HTTP Server, Windows Server
SDK / API and Integrations	ASP.NET Web API, RESTful API
Virtualization, Containers and Orchestration	Docker, Kubernetes

Version Control	Git, GitLab
Logging and Monitoring	Grafana
Mail / Network Protocols / Data transfer	HTTP
Deployment, CI/CD & Administration	Jenkins
Message/Queue/Task Brokers	Kafka, RabbitMQ
Other Technical Skills	Adobe InDesign Server, App Metrics, Kilgray memoQ Server, Microsoft Servers, Microsoft Service Bus, Microsoft System Center Operations Manager, Multithreading, Parsing, Rendering, SDL Trados Studio, Windows Installer, Wix Installer

EXPERIENCE

Principal Software Engineer, Technical Lead, Uklon

September 2021 – Now

Description: Uklon is the number one ride-hailing solution in Ukraine, founded almost simultaneously with Uber, serving 100,000 orders per day. Technologically, it is an R&D product company, with Rider and Driver mobile apps and 100+ micro-services of the backend.

I was hired to build a new delivery solution from scratch, including last-mile and cargo delivery. First of all, it was a backend orchestrator that managed the order flow with the following non-functional requirements: 100 orders/minute with 20000 active drivers under normal conditions, up to 1000 orders/minute at peaks. I delivered that. The initial MVP release was after 6 months and 10 more releases since then. The main technical goals were high availability and compatibility.

Kafka was used extensively, for asynchronous communication, for breaking ladder dependencies, and for caches, all core operations were processed asynchronously and were idempotent. In some places, payments specifically, exactly one semantic needed to be achieved. Some operations were synchronous and delivered through HTTP.

When designing features, I was working closely with the solution architect and with 7 other teams, considering their implementations and needs. Usually, I presented cross-service contracts to them before implementation. My team included backend developers as well as mobile developers for Android and iOS apps.

Technologies and platform: C#, .NET Core 6, Apache Kafka, PostgreSQL, HTTP RESTful API, Hangfire, Docker, Kubernetes, Grafana, GitLab

Senior Technical Lead, DraftKings (NASDAQ: DKNQ)

November 2019 – August 2021

Description: DraftKings is an American daily fantasy sports contest and sports betting operator. I came to the company to gather a new team, take over the new version of the



Betting API, deliver it to production to meet certain non-functional requirements, and run after that.

The Betting API is a set of highload microservices to calculate and place bets. It accepts requests via HTTP, also delivering delayed results through SSE. Non-functional requests were: 100 ms for calculation, 1 s for placement, and 50000 RPM. The solution has over 30 other HTTP and Kafka stream dependencies (managed by other teams in the company). One huge dependency is sports data, coming with a speed of up to 10 mil messages per minute through Kafka.

My main responsibility was the project's technical design, the meeting was given non-functional requirements and general technical excellence. I worked closely with my team of 6 engineers, doing also people management, and being their director.

After taking over the project, I wrote a high-level technical roadmap to improve the maintainability, scalability, and reliability of the project. For the next year, I had been perusing the roadmap by decomposing it into technical epics and tasks, writing acceptance criteria, scheduling the tasks, and reviewing the results.

Technologies and platform: C#, .NET Core, TPL DataFlow, Apache Kafka, Aerospike, Docker, Kubernetes, Grafana, Kibana, App Metrics, Rabbit MQ, Git, Jenkins

Senior software engineer, Altium

May 2019 – October 2019

Description: Altium provides electronics design software. It has been developing its software for 35 years and has a huge 40 GB monorepo with C#, C++, and Delphi code. I was part of its R&D team (which was pretty flat), delivering a few isolated projects in C#, familiarizing myself with their ecosystem, and aiming for their cloud platform. The most interesting ones were:

- writing a Git integration provider over Git CLI for their CAD solution.
- migrating a bunch of full-text indices (around 10 GB of data) to ElasticSearch in Amazon.

Technologies and platform: Windows Server, .NET Framework, .NET Core, Amazon AWS, ElasticSearch, Git.

Software Architect of the Translation Tools team, LanguageWire

Overall: 6.5 years

May 2017 – April 2019 Software architect (2 years)

January 2014 – April 2017 Software technical lead (3.5 years)

Feb 2013 – December 2013 Software engineer (1 year)

Description: LanguageWire is a translation agency with its own software platform. I came to the existing development team as a software engineer. Over time, I naturally took the responsibility for a bunch of existing services, created several new ones, and became a software architect, responsible for 8 products in the Translation Tools team. I was running the full application lifecycle: design, development, testing, releasing, and post-production.



Specializations there were:

- writing micro-services that were parsing, processing, and storing relatively big documents in popular formats (HTML, Word, InDesign, etc.), processing huge documents like exported translation memories (up to 1 GB of XML files), indexing, etc.
- working with closed formats like Adobe InDesign, SDL Trados Studio packages
- designing a new generation of the public API for corporate clients, maintaining compatibility, and client libraries for different languages (C#, Java, PHP).
- developing a new computer-assisted tool for translators.

The things that I have done across services to move them to the next level:

- Eliminated all unnecessary dependencies between services, set up boundaries, turned some to event-based to ensure one service degradation will not kill the entire platform and separated ones that historically had but should not have a shared database. This allowed them to turn them into real micro-services.
- Set up and enforced the development pipeline for all existing and new projects. This includes continuous integration and the release pipeline.
- Covered products with infrastructural and operational monitoring.
- Worked with the IT department on building up a private cloud for the platform
- Wrote a distributed scheduler on top of Microsoft Service Bus to execute long-term tasks across multiple instances.

Technologies and platforms: Windows Server, .NET Framework, SQL Server, Microsoft Service Bus, ASP.NET Core, ASP.NET Web API, Wix Installer, Microsoft System Center Operations Manager, Adobe InDesign Server, Kilgray memoQ Server, SDL Trados Studio, TypeScript, Knockout.

.NET and C++ Developer, Datalogics

January 2012 – January 2013

Description: I was working with the source code of an Adobe library for PDF rendering. It was a cross-platform library written in C++ with some platform-specific code. I was creating a set of COM interfaces for Windows on top of the library, a C# wrapper, and a sample Windows desktop app. Later I was porting the library to Windows Runtime and to ARM and x64 architectures, so it could run on devices like Surface and Windows Phone.

.NET Developer, Ximble

June 2011 – December 2011

Description: I developed a .NET application in a small team, and had daily communication with a US customer.

Responsibilities: My responsibilities were both back-end and front-end development, there was no strict specialization. Back-end responsibilities included working with the database schema, creating ORM mappings, developing a business layer, writing LINQ queries and adjusting them using an SQL profiler, and developing a web UI layer. Front-end tasks were html mockup integration and writing Ajax scripts.



My own “garage” project

2004-2006

Description: I was writing it on my own, and even sold several “licenses”. It was an application for knowledge testing. Tests of various kinds could be composed. There was a user façade as well as an administrative panel to create tasks, control the process, and view results. Data were stored on a server in a Firebird database.

Technologies: Delphi 7, Delphi 11, C++, Firebird, HTML, Windows Installer

EDUCATION

Bachelor of computer engineering degree

University: Donetsk national technical university

Faculty: Computer Science and Technology

Major: System programming

Period: 2006 – 2010

Average grade: 4.7 / 5.0

Master of computer engineering degree (without a degree)

Period: 2010 – 2012

AWARDS

All-Ukrainian Student Programming Contest: system programming

2011

All-Ukrainian Student Programming Contest: computer engineering.

2010

