Last updated on August 2, 2021

# Kirill Timofeev


| | | | |
|---|---|---|---|
| Email: | cv@kruzenshtern.org | Location: | New York, US |
| Homepage: | https://kruzenshtern.org | Status: | Permanent resident |


## Summary


A principal architect & a technical leader with over 15 years of professional expertise in the IT industry, consulting and management.

Experience in delivering enterprise-grade projects for some of the world's largest institutions in financial services and capital markets such as securities settlement platform, a digital money solution for payments and FX, and trading systems for an investment bank.

Excellent polyglot full-stack development skills. Fluent in many programming languages including Python, Ruby, Javascript/Node.JS, and Lua.

Personal projects spans from building educational tools for aspiring developers, hacking of IoT devices to architecting low-latency systems.


## Technical skills

| | |
|---|---|
| **Python** | I have a deep experience with Python 3. I built financial solutions that connect quants, data analysts and back-office teams together to process data in real-time rather than at the end of the trading day. |
| **Ruby** | I felt in love with Ruby after I read Why's (Poignant) Guide to Ruby back in 2006. Since then I've been using Ruby to build a wide range of applications such as a web-crawling system, web applications on Rails, a cloud-native background job manager and many others. |
| **Javascript Node.JS** | I have a heartfelt experience with Javascript development over the last decade. I was lucky enough to see Javascript and Node.JS evolution over this peroid, i.e. I have used vanilla Javascript, Prototype/jQuery, Backbone, Angular, Vue, Ember.JS, React, and Svelte frameworks along with architectural patterns such as reactive programming and saga. |
| **Other skills** | I would be happy to discuss my commercial experience with and exposure to Kubernetes, Nomad stack, Terraform, AWS services, serverless architecture, Hyperledger, Ethereum/Solidity, WebAssembly, Firecracker, building a CI/CD process on Jenkins, Concourse and Bazel. |

## Employment

| | |
|---|---|
| 2019–today | **Principal Architect / VP**. DataArt Solutions Inc. |
| 2015–2019 | **Software Project Manager**. Led a blockchain domain at DataArt with a focus on the finance industry. Explored blockchain technologies in other domains including travel, healthcare and life sciences, insurance, and IoT. DataArt Solutions Inc. |
| 2010–2015 | **Solutions Architect**. DataArt Solutions Inc. |
| 2008–2010 | **Team Lead**. DataArt Solutions Inc. |
| 2007–2008 | **Senior Developer**. DataArt Solutions Inc. |
| 2003–2007 | **Software Developer**. Izhevsk State Technical University (ISTU) |

## Education

I received a Master's degree in Mathematics and Computer Science from St. Petersburg State University of Information Technologies, Mechanics and Optics (ITMO), Russia in 2009. During the second year of Ph.D, I was offered a promotion at DataArt and focused on business projects instead.

## Recent Projects

2019–today: **A Trading Platform for an Investment Bank**
**Position**: Principal architect & technical leader
**Team**: 30 engineers
**Technologies**: Python, Javascript/React, Java, Kafka, Kubernetes, AWS services, MongoDB, Postgres, Airflow, Bazel
**Summary**: A client is building a new trading platform that brings the entire trading process online, eliminating the need for phone calls and emails between investors and traders. Surveillance, risk management, and reconciliation happen automatically and in real-time rather than at the end of the trading day, with customized data flows between relevant departments. The platform enhances transparency, expands trading capacity, improves alpha generation, and helps traders manage risk more effectively. The platform uses modern, composable data architecture, sourcing real-time data from a variety of providers and exchanges. The range of supported assets is continuously expanding and currently includes mortgages, corporate loans, high-risk construction loans, digital currencies, and traditional securities. Engineered to automate tasks such as gathering and normalization of data and generation of analytics, the platform allows data scientists to focus on algorithm design instead of data processing. The platform was designed to be cloud-agnostic, a client can easily license its technology to institutional investors, asset managers, hedge funds, and proprietary trading firms.

2018–2019: **Private Capital Markets Platform**
**Position**: Principal architect & technical leader
**Team**: 10 engineers
**Technologies**: Ruby, .NET, Javascript/React, Amazon Lambda and other AWS services
**Summary**: The project aimed to develop a private capital markets platform that connects investors to private investment opportunities in both companies and funds. The client wanted to upgrade the existing platform to the next version as well as to align it with the best practices in cloud, security, regulatory compliance and deployment. One of the challenges was to migrate the application, processes and infrastructure to a cloud-native development (lift and shift was not an option.) It included data migration strategy, a monolith application split up into manageable and shareable services, and design and development of new deployment pipelines as well as a cloud infrastructure development.

2017–2018: **Alternatives Market Platform**
**Position**: Principal architect & technical leader
**Team**: 7 engineers
**Technologies**: Ruby, Javascript/Ember.JS, MySQL
**Summary**: The project aimed to create a uniform platform that streamlines private transactions and allows qualified investors and institutional buyers to proactively identify, review, and confidentially engage in private transactions across different industries. By creating a single forum for the private capital markets, the platform delivers unparalleled access, transparency, and efficiency to investors, broker-dealers, and issuers. The platform development was aligned with the design and creation of a cloud infrastructure within a highly regulated financial environment. To streamline the development the automated release management strategy was developed. It allowed to deploy and scale new releases to cloud instances almost instantaneously.

2016–2017: **Distributed Digital Ledger Platform**
**Position**: Principal architect & technical leader
**Team**: 6 engineers
**Technologies**: Ruby, Javascript/Ember.JS/D3, MySQL, DLT technologies
**Summary**: The project goal was to develop a solution that allowed to trade assets digitally using a blockchain-based platform (digital money and FX transactions.) It provided a digital ledger technology to facilitate the issuance and recording of transfers of shares of privately held companies. The solution used cloud-based capitalization table management and stock plan administration solutions along with distributed ledger technologies to secure private transactions. To streamline coordination and integration between three finance companies, API-first approach was chosen to develop a money settlement platform. Team composition allowed breaking down the platform into smaller services, e.g. blockchain communication/reconciliation, authentication (JWT and OAuth 2.0), settlement and others. API endpoints, their responses and error messages were documented using Swagger and supported by integration test suites that could be easily executed to validate each of those applications in isolation and in real-life conditions.

# Personal Projects

**Diggy – a zero-setup development environment**
**URL**: `https://diggy.sh`
**Technologies**: Javascript, Python, WASM, Ruby, Terraform, AWS, NsJail
**Summary**: Diggy is an open source out-of-the-box solution for aspiring developers that helps them start working on a project in less than a second with no overhead in setup and deployment. It is Google Docs, but for coding. The grand vision is to make coding a magnitude easier. I am working on the next iteration after an interview with YC for S21 batch.

**Maze – a Pico-8 game**
**URL**: `https://github.com/oneearedrabbit/maze`
**Technologies**: Pico-8, Lua
**Summary**: A game where a protagonist explores a maze to find its secrets. A solo-development to explore game development mechanics and principles. A game engine is built in Lua. Also, I tried to keep retro aesthetics by building everything including a lighting system in 1-bit palette, which comes with its own challenges.

**A low-latency trading solution**
**URL**: `https://github.com/oneearedrabbit/ring`
**Technologies**: Go, Python, lock-free data structures
**Summary**: Developing of ultra-low latency algorithms that fetch market data and do statistical computations. I've been using lock-free data structures that allow to process a message in less than 20ns, which gives the performance of 50mln messages per second on a single machine, which is enough to process L2 market data for US equities. The system is relatively easy to scale horizontally, therefore throughput could be increased even more if needed.

**A small-scale social network**
**Technologies**: Node.JS, Javascript/Ember.JS, Redis, Docker
**URL**: `https://github.com/pepyatka`
**Summary**: Pepyatka is an open-source FriendFeed clone, the real-time aggregator and social network. Basically, it is a small-scale social real-time feed aggregator that allows one to share cute kitty photos, coordinate upcoming events, discuss any other cool stuff on the Internet. FriendFeed was shat down by Facebook in 2015, and Pepyatka was aiming to recreate the core functionality for a small and active community that was using FriendFeed at that time. Later, Pepyatka has been taken over by **FreeFeed** team, which continued design and development of the platform.

**Reverse engineering of an IoT device**
**URL**: `https://github.com/oneearedrabbit/karotz`
**Technologies**: Wireshark and a bit of luck
**Summary**: Karotz is a Wi-Fi enabled device that was originally manufactured by Mindscape company. It uses a closed-source infrastructure to communicate with. After Mindscape filled for bankruptcy they discontinued web services making Karotz a cute, but useless device. I reverse-engineered Karotz's protocol and figured out how to get root access to the device, which ultimately allowed me to build my own infrastructure to prolong Kartoz's life. Since then it was picked up by other hackers.

# Recent publications

In clinical trials, the future lies in blockchain technologies. Apr 2019. Deutsche Zeitschrift fr Klinische Forschung. `https://www.dataart.com/downloads/dataart_dzfk_k_timofeev.pdf`

Supply Chain in the Flow. Oct 2, 2018. `https://www.chemanager-online.com/themen/strategie/supply-chain-im-fluss`

Blockchain and Healthcare. Nov 2, 2018. `https://www.hcm-magazin.de/blockchain-und-healthcare/150/10668/379127`

How blockchain networks could radically empower patients. Jan 2, 2019. `https://digitalhealthage.com/how-blockchain-networks-could-radically-empower-patients/`

Building a blockchain MVP. Oct 6, 2017. `http://www.bobsguide.com/guide/news/2017/Oct/6/building-a-blockchain-mvp/`