Aleksandr Vinokurov

+31(6)13018394
aleksandr.vin@gmail.com
linkedin.com/in/aleksandrvin
github.com/aleksandr-vin
https://aleksandr.vin

Team Lead & Senior Backend Developer

Skills

Leadership / Team skills

- Resourcefulness and creativity
- Creating enthusiasm in team
- Mentoring teammates
- Driving team toward the goal
- Establishing safe environment for experiments

Software Development Skills

Types of Programming

Functional, Declarative, Reactive, Imperative, Object-Oriented

System Design

Cloud-based, Serverless Structures, Small Independent Services (Microservices), Handling Multiple Tasks at Once, Event Systems, Reliable & Expandable Designs, Managing User Loads, Organizing Data & Services

Service Creation

Web Service Designs, Traditional (RESTful) & Modern (GraphQL) APIs, Real-time Connections (Websockets)

Work Approaches

Agile Processes, Scrum Framework, DevOps Collaboration

Quality Assurance

Test-Driven Development (TDD), Behavior-Driven Development (BDD), Split User Testing (A/B), Thorough System Checks, Performance Tests

Tech Knowledge

Regular, Time-based & Flexible Databases; Custom Scripting (DSL)

Platforms & Tools

Cloud Platforms, Software Containers (like Kubernetes), Focus on AWS

Security Measures

Checking for Weak Points, Data Protection (Cryptography), Ensuring Safe Data Transfers

Additional Skills

Automating Tasks, ChatGPT/OpenAI Integration, Continuous Development & Testing (CI/CD), Diagnosing Issues, Software Speed Improvements, Merging Systems Together

DevOps Skills

”Automate everything”, ”Configuration-as-a-Code”, ”Measure everything”, Docker, Kubernetes, CI/CD, AWS, GitHub, GitLab, Ansible, Bots automation, Event monitoring systems (Prometheus, Grafana, Kibana)

Programming Languages

Scala / Akka, Python, Bash, TypeScript, JavaScript, Erlang/OTP, Node.js, C, C++, Rust, Go, Haskell, Lisp, Java, Regex, SQL, XML/XSD/XSLT, Ansible

DSL (Domain Specific Languages)

EBNF grammar design and implementation

Networking

OSI layers from 2 (Ethernet frames) to 7 (HTTP, FTP, SMTP)

NoSQL, RDBMS and Expert Systems

Elastic Stack, HBase, Postgres, Redis, Couchbase, DETS (Erlang/OTP), Oracle, CLIPS, Prolog

Event, Message and Task Queues

Kafka, Redis Queue, ZeroMQ, Celery, Beanstalkd

Infrastructure

- Helm, Kubernetes, Containers, Serverless
- AWS: Lambda, S3, EC2, Timestream
- Consul, Zookeeper, Vault
- HAProxy, NGiNX, Kong Gateway, DNS Load Balancing

Build / Source Control / Collaboration

Git, GNU Autotools, rpmbuild, GitHub Actions, GitLab CI, Jenkins, Azure DevOps, Bamboo, JIRA

Mobile App Development Skills

- Full life cycle of app development for iOS and Android platforms
- Languages: Objective C, Swift, Java
- Services: Location Services, Bluetooth Low Energy
- Frameworks: Reactive Cocoa Framework, Mapbox, RestKit, KIF, Google Analytics, Twitter Fabric, Intercom, New Relic
- UI/UX: Sketch, Flinto, Zeplin

VR/3D Development Skills

Unity, Blender, Oculus SDK

IDE and Tools

- Visual Studio Code, Emacs, IntelliJ IDEA, Xcode, Android Studio
- Docker Compose
- Numerous Unix hacker's tools (gdb, grep, sed, awk, dd, expect, tmux, screen, cron, strace, dtrace, tcpdump/wireshark, ipset/iptables, nc, socat, etc.)
- macOS automation (AppleScript, have my Safari Extension)
- ChatGPT consultations

OS

Hacker/Administration skills in Linux Servers, macOS

Security

Networking and Web security, Authentication-Authorization/OAuth2.0, JWT, Cryptography/Encryption, Analytics, Social Engineering, Penetration Testing

Certificates

- Unity Essentials Pathway, Issuing authority Unity Technologies, **January 2021**
- Improving Deep Neural Networks: Hyperparameter tuning, Regularization and Optimization, Issuing authority Coursera, **June 2018**
- Functional Programming Principles in Scala, Issuing authority Coursera, **July 2016**
- Agile Development With Scrum, Issuing authority ScrumTrek, **April 2011**

Education

2000-2006

Technical Cybernetics, Specialist, St.Petersburg State Polytechnic University

2014 Spring-Summer

Graduated, Founder Institute, fi.co

Miscellaneous

- **Languages:** Russian (native), English (fluent), Dutch (A2)
- **Citizenships:** Netherlands, Russian Federation
- **Work permit:** In the Netherlands
- **Location:** The Netherlands

Work Experience

August 2021–Now

Senior Backend Developer, Team Leader at the Operation Portal team at KLM, klm.nl, Amsterdam, The Netherlands

My goal was to design and develop new features of Operation Portal. Mentor junior developers. Organise standby support and educate the team. Manage project instrumentation and optimization, make it more robust. Develop database schema migration scripts. And lastly, curate the process of migration to the Cloud.

Product was started as a proof-of-concept at the beginning of the Corona crisis and was built with Scala and TypeScript on top of HBase, GraphQL and React. It supports flights and passengers scheduling operations with close to real time data updates.

Most of my contribution was to portal's backend, but I also have created many pull-requests to our frontend and different other teams' projects, working with Scala and Python.

During my work at KLM, I designed and implemented multiple developer tools and CI/CD pipelines:

- Security checks pipelines
- Public Docker images adoption framework
- Grafana dashboards-as-a-code system github.com/aleksandr-vin/grafana-dashboards-provisioning
- Created proof-of-concept of a scalable REST API service for proprietary costs calculation library
- Created KLM Abbreviations Dictionary (an Apple Dictionary)
- Created VR application of Operations Control Center building for proof-of-concept of new information screens (in Blender, Unity and Oculus Quest 2)

August 2020–August 2021

Senior Backend Developer of the Streaming Data Analytics platform at ING Nederland, ing.nl, Amsterdam, The Netherlands

I joined the company as Senior Backend Developer in August 2020 with the goal to help an existing team develop their Streaming Data Analytics platform, which is based on Apache Flink and Ververica Platform. The platform provides real time ML scoring service and allows tenants write their logic in DSL.

November 2019–July 2020

Team Lead at StackState, stackstate.com, Utrecht, The Netherlands

StackState is a product company of Xebia Group. It was founded in 2015. I joined the company as Senior Backend Developer in May 2019 with the goal to take Team Leader's role in 6 months. In September 2019 StackState was named a Cool Vendor by Gartner.

After initial 6 months of work in the company I took the Team Leader role of feature delivering team of 6 people, including me and a Scrum Master. Two of the developers were working remotely, one in the -6 hours time zone. During these initial months I set up 1:1 calls with all team members, took initiative and proved to be a support and mentor for the team. During the whole period of work in StackState I was mentored by my manager and held 1:1 meetings with him on a bi-weekly basis. With the strike of corona-crisis we transformed our process to completely remote-working one, while still being able to successfully deliver on the goals that team committed.

My responsibilities were: organization and maintenance of the team, team focus on deliverables and goals, mentoring team-members, spotting and resolving team problems and escalating when needed.

At the end of June the company made a decision to cut down costs and close the team, letting us go.

I was participating in delivering 4 major releases of the product.

With the Team Leader's role I was also having a Release Manager's role to organize, control and teach other developers of the release process, also to prioritize improvements to that process automation.

May 2019–July 2020

Senior Backend Developer at StackState, stackstate.com, Utrecht, The Netherlands

I started with the feature delivering team, whom I was planning to lead eventually. We delivered some of the core functionalities of the company's product, like Telemetry Perspective feature and StackPack (proprietary packages) upgrading feature. And we were participating in the short-term development of Freemium plan's support features.

During first months I took ownership of the release process, formalised and automated it, developed a tool to operate the release process pipeline (put it on GitLab CI) and integrated it with JIRA, Slack, AWS, JFrog Artifactory, GitLab Pipelines, Quay.io and other third-party services.

Separately from features development I performed multiple optimizations to our GitLab pipelines, proposed new erratic tests handling process, several proof-of-concepts from automation of hotfix rollouts to optimizing our product Docker images.

- **Languages:** Scala, Python, Bash, TypeScript, Java
- **Frameworks/Libraries:** ZIO, Akka Streams, Akka HTTP, Websockets, Avro, Protobuf, ScalaTest, ScalaCheck, Mockito, JMH and ScalaMeter, proprietary code generators, SBT plugins
- **Services:** Elasticsearch, TinkerPop, Kafka, Splunk, AWS, Docker

August 2016–April 2019

Senior Backend Developer at Veon, veon.com, Amsterdam, The Netherlands

Veon is a big multinational company, formerly known as Vimpelcom, that has different telcos across the globe. I joined the company in its journey of digital transformation, helping to develop Engagement Platform with the goal to become a content provider as well as a service provider.

Last project (mid 2018–beg 2019)

Leading Core Experience, Identity & Safety team of 5 backend developers (including myself), 4 Android and 4 iOS developers (partially remote). Along with the support of onboarding, authentication, management of the platform users and their consents, the team is busy with remov-

ing/refactoring legacy backbone code that was built at the very beginning of the platform by a third party. That required coordinated work across all the teams and proper planning and execution on the running platform. As a side project for part of the team, we are designing and prototyping a generic search service to support full text search in different aspects of the platform. The engine is Elasticsearch and Kafka. We are using the same common stack of technologies that we already used in previous projects in the company.

Third project (end 2017–mid 2018)

Designing, estimating and developing Consents Management Service with mobile SDK. In scope of GDPR adoption, I designed a service to manage consents versions and track users' decisions. Service supports different locales of a consent text, filtering by user's attributes, versioning of legal meaning of the consent and smart invalidation of user's decisions. Service utilizes HTTP caching to reduce client requests to a reasonable amount. Uses Akka Streams to perform batch import/export operations with failovers. Supports event-driven operations. To implement the service I gathered and led a small team of 1 more backend and 2 mobile app developers. We were using the same stack of technologies as in previous projects.

Second project (mid 2017–end 2017)

Participation in Platform team. Our goal was to optimize and develop a common layer of our microservices, working closely with our infrastructure team. We moved from heterogeneous to homogeneous Akka clusters with sharding, and introduced event driven architecture with Kafka. We applied monitoring of our services with the help of Kamon framework and developed Grafana dashboards. For benchmarking we used Gatling and JMeter as well as Java Microbenchmark Harness, Elasticsearch and Python. During this project, while working close to the infrastructure team, I developed some automation and monitoring command line tools using Python, Bash, Scala scripting with Ammonite, ssh tunneling, docker and psql util.

- **Languages:** Scala, Python, Bash
- **Frameworks/Libraries:** Akka Actors, Akka HTTP, Akka Streams, Akka Cluster, Kamon, Slick, Flyway Migrations, Protobuf, ConstructR, ScalaTest, ScalaCheck, ScalaMock, Mockito, Swagger
- **Services:** Kafka, Elasticsearch, Consul, Kong API Gateway

First project (mid 2016–mid 2017)

Developing Identity Management services from scratch, responsible for onboarding, authentication and managing of the users. Initially the flow was supported by third party services, and our goal was to move the backend support to our site in steps that required API design with mobile app developers, integration across AWS locations and design of appropriate strategy to execute the migration. Also we set up Elasticsearch stack at that time and embraced structured logging, brought unit testing and set up integration testing with the help of Docker Compose. Applied code review process in our merge requests. Frankly speaking, we've set up the team culture.

- **Languages:** Scala, Python, Bash
- **Frameworks/Libraries:** Akka Actors, Akka HTTP, Akka Cluster, Slick, Flyway Migrations, Scalaz, Argonaut, Spray, ScalaTest, ScalaCheck, ScalaMock, Mockito, Swagger

October 2007–July 2016

Senior Software Engineer at Peter-Service, CJSC, billing.ru, Saint-Petersburg, Russia

Peter-Service is a big software company (was about 1000 employees across Russia), it develops and supports billing systems for telco.

Last project (mid 2015–mid 2016)

Full-stack logging infrastructure for distributed architecture. It solves two problems: (1) aggregating logs from distributed services in one place with indexed full-text search and (2) runtime controlling of the produced log detalization level for specific request-id's/user-logins/sessions on each node of the distributed service. The solution consists of (logs) Aggregation infrastructure, (logs) Analytics service and distributed services Dynamic Log Control service. Aggregation and Analytics are based on Elastic Stack — Elasticsearch-Logstash-Kibana. Dynamic Log Control service was designed as pluggable agents for Talend ESB and Tomcat with RESTful API microservice that synchronize agents via Zookeeper. Allowing logging of all operations that were originated by specific user sessions across all servers, and have that logs being produced in DEBUG level with minimal effort and overhead. Distributed deployment was implemented in Ansible.

- **Languages:** Java, Ansible
- **Frameworks/Libraries:** Swagger, Logback, Jackson, OSGi Pax Logging, Apache Curator
- **Services:** Elasticsearch, Zookeeper, Logstash, Filebeat, Kibana

Third project (end 2012–mid 2015)

Distributed high-availability system for providing traffic rate-limiting/shaping and Level 7 DDoS protection (based on HAProxy, designed solely from scratch) with dynamic rules in DSL, dynamic reconfiguration by Zookeeper and a RESTful JSON API. The main design goal was to develop a scalable solution that can be run on a farm of low-level machines and when capacities can be scaled on demand. This goal was achieved mostly by using a "shared-nothing" model and by implementing performance-critical parts in C language using HAProxy as a base engine.

HAProxy is a high-performance open source TCP and HTTP load balancer written completely in pure C. It has a threadless "shared-nothing" design. I've done analysis of its sources, designed the architecture of extension that was needed for our appliance and implemented it with unit- and performance tests, checking the result with profiler. While embedding the CLIPS (C Language Integrated Production System) into the HAProxy, I was using Valgrind to find the memory leak that appeared under load, finally I've discovered that it was an expected behaviour of the garbage collector, specifically implemented in CLIPS, and I was able to workaround the leak.

- **Languages:** C, Python, Java, JavaScript, CLIPS
- **Frameworks/Libraries:** Protobuf, ZeroMQ, Python PLY, Python Mock, CMocka
- **Tools:** Emacs, Strace, GDB, Bash, cURL, JIRA, Confluence, GIT, Rpmbuild, Valgrind, GNU Autotools, ipset/iptables, RRDtool, Icinga
- **Services:** Couchbase, Zookeeper, CLIPS
- **Web:** Angular JS, Java EE, Spring, Camel

Second project (beg 2012–end 2012)

Development of the infrastructure for new services of the company. Main requirements were: high availability and scalability. From April to August, 2012, I've implemented an OAuth v2 specification in an Erlang server (as a part of the SSO solution).

During my experience with the Erlang/OTP language and environment I've experimented with the open source logging framework and proposed a pull request with asynchronous mode for this system.

- **Languages:** Erlang/OTP
- **Frameworks/Libraries:** Erlang/OTP, Cowboy, Common Tests, ibrowse, SSL
- **Tools:** Emacs, Bash, Rebar, cURL, Git, Hg
- **Databases:** Mnesia, DETS/ETS

First project (mid 2007–end 2011)

CRM-CMS system for mobile operators, https://web.archive.org/web/20150810163328/http://www.billing.ru:80/solutions/customer.

My participation in the project:

- Porting backend codebase from Windows to Red Hat Enterprise Linux, HP-UX and Sun Solaris servers
- Designing and implementing of a plugins' API for portable backend system
- Refactoring of the Business-Objects layer
- Debugging and profiling Unix-related issues of the backend

In this project I mostly worked on the backend. Its codebase was written in C++ and consisted of Business-Objects layer, Common Utilities and Business Logic. My first task was to port code base from Windows platform to Unix-derived ones. I was the only Unix expert in the team during my work on the project and I've educated the team that at the time of my leave they were able to support the codebase on their own. Business-Objects layer consisted of about 400-500 classes and extensively used templating. One day we ran out of IDs in the link-phase under HP-UX. After that issue my teammate and I have done the refactoring of the whole Business-Objects layer and reduced compilation time from 50 min to slightly more than 20 min.

Another task was to design and implement a plugins system for the Business Logic layer to be able to call pluggable modules in the CRM workflow.

As I was the only Unix expert in the team, I was working on all the Unix-related issues of the backend. I've done excessive Valgrind and GDB runs to find memory leaks, heap misuses, race conditions and deadlocks as well as to solve global constructors issues of dynamically loaded libraries. I've used gprof to periodically find bottlenecks in libraries. And used to run dtrace for run-time tracing under Solaris.

During my work on this project I've designed two DSLs (in C++ templates and macros): for embedding XML documents and SQL queries in C++ code with preprocessor-time checks and compilation, https://medium.com/@aleksandrvinokurov/dsl-for-xml-in-c-f284453819b

I've started automation processes in the team: from automation of the build and release process to nightly builds and unit-tests.

- **Languages:** C++, SQL, PL/SQL, XML, XSLT
- **Frameworks/Libraries:** OTL, Boost Library, Xalan-C, Xerces-C, Loki, Boost Testing Framework, Google Test Framework, Google Mocking Framework, Log4cxx
- **Tools:** Emacs, GDB, DTrace, GCC, GNU Autotools, Bash, Clear Case, Clear Quest
- **Databases:** Oracle 11g

April 2006–October 2007

Software Engineer at Alarity Corporation, Saint-Petersburg, Russia

Alarity Corporation was a medium company of about 40 people, developed video codecs and outsourced the development of the VPN solution for Netilla corp (was acquired by AEP). Colleagues were located in New Jersey and communication was by voice and emails.

My participation in the project:

- CPU & network usage investigation of different compression libraries
- Design and implementation of a load balancing architecture for VPN solution
- Design and development of a serial console for servers' configuration

March 2005–December 2005

Security Auditor and Software Developer at Digital Security, dsec.ru, Saint-Petersburg, Russia https://web.archive.org/web/20051219054309/http://www.dsec.ru/

Digital Security at the time I was employed was a small company (about 10 employees), performed security audits for big national industry companies and developed their brand risk assessment product.

The security team was responsible for onsite technical security audits and penetration tests. There were occasionally "in-field" development of networking tools, in C and Perl, for security audit support.

My participation in the project:

- Developing a part of the risk assessment product's core
- Designing and initial development of a programmable framework for building scanners for network service's vulnerabilities

April 2004–March 2005

Software Engineer at Center-Inform, FGUP, center-inform.ru, Saint-Petersburg, Russia

Center-Inform was a medium sized company, developed information systems for process automation.

My participation in the project:

- Transact-SQL module for searching of the typos in words in database records
- Proprietary cryptographic library
- Drivers for barcode printers
- GSM modem driver

Startup Experience

Spring 2014

From Spring of 2014 I was a co-founder & CEO of Evecon Corp. — startup, https://web.archive.org/web/20160304170331/http://evecon.co/

We started developing mobile connectivity and indoor navigation service but later pivoted to development of online service for eventing market (was attendlist.co). My co-founder & I graduated from the Founder Institute, fi.co in summer 2014. While we were bootstrapping, I was the only one backend developer where I've used Scala, Go, Python, Twitter Bootstrap, Angular JS for frontend and backend and Swift/Objective-C/Java for mobile.

Winter 2015

In January 2015 we released our first iPhone app on App Store: BizCardApp. You can watch a demonstration video here: https://youtu.be/TJJCMxzSrDI for our landing page of that time. Its REST API backend and web server has test stage and production stage servers with automatic deployment process (based on git post-receive hook). I've used this stack: Nginx reverse proxy, Supervisord for supervising servers, Gunicorn as WSGI server, Flask web framework, SQLAlchemy as ORM, Alembic for DB schema migrations, Jinja2 templating system, WTForms for POST forms and Twitter Bootstrap for frontend design.

Summer 2015

During summer of 2015 we made another app for the local Harley Davidson event, called St. Petersburg Harley Days, https://www.linkedin.com/company/st-petersburg-harley®-days/about/.

You can find it on App Store https://itunes.apple.com/us/app/harley-days/id998388857?mt=8 and Google Play https://play.google.com/store/apps/details?id=co.evecon.hdprototype. The main achievement of that project was a very short time to market: we made it in 1.5 months, working in our spare time and cutting features' scope for the goal of meeting the deadline. At the end we promoted the app during event days to collect installations, and registered only one known crash for iOS 7. The UI and UX design is not of its best, but the architecture allowed us to be busy updating only the content for every event up to recent days.

2016

We released another mobile app for iOS and Android to engage Harley Owners in Russia and CIS: https://itunes.apple.com/us/app/hog-plus/id1434462365?mt=8. The app was designed and implemented for H.O.G. Russia & CIS. Apart from the app we developed a back-office to manage loyalty programs and events in the app.

2020

During Winter of 2020-2021 we developed a live parking detection system, that was using public street web cameras to collect photos of parking areas in a city, upload them to S3 buckets, crop photos to mapped zones on AWS Lamba, detect cars there with YOLOv4 https://github.com/videoparking/darknet and store results in AWS Timestream. Service was able to say how full the parking area was at the moment (based on history data) and was coloring zones on a map. We freezed the project in the Spring of 2021. Public components can be viewed here https://github.com/videoparking.

Personal Projects

- ChatGPT assistant for Marktplaats, a Telegram Bot
  https://github.com/aleksandr-vin/marktplaats-gpt
- CLI client and Prometheus exporter for mijn.simpel.nl
  https://github.com/aleksandr-vin/mijn-simpel-exporter
- ChatGPT Pair Programming, a VSCode Extension
  https://marketplace.visualstudio.com/items?itemName=aleksandrvin.chatgpt-pair-programming
- SBT Keychain Credentials Plugin
  https://github.com/aleksandr-vin/sbt-keychain-credentials

Opensource Experience

I contribute to the open source projects that I use — you can check my GitHub profile: https://github.com/aleksandr-vin.

Contact Information:

Phone:	+31(6)13018394
Email:	aleksandr.vin@gmail.com
LinkedIn:	linkedin.com/in/aleksandrvin
GitHub:	github.com/aleksandr-vin