ABOUT

Yaroslav (Slava) Tkachenko, Software engineer

I started programming when I was 13, professionally since 17. Programming was always my passion and I never stop learning.

During my career I did a lot of different things, from slicing mockups to HTML/CSS and writing single page web apps to designing and building microservices architecture and implementing complex back-end services with Actor-based frameworks.

My current areas of interest:

- Distributed systems and microservices
- Back-end programming
- Infrastructure, DevOps tools and practices

You can find my experience, education and skills in the pages below.

CONTACTS

2108 - 999 Seymour street, V6B 0M5, Vancouver, British Columbia, Canada.

- Phone: 604-230-8734
- Email: sapiensy@gmail.com
- Website/blog: sap1ens.com
- Twitter: twitter.com/sap1ens
- GitHub: github.com/sap1ens
- LinkedIn: ca.linkedin.com/in/sap1ens
- StackOverflow: stackoverflow.com/users/899937/sap1ens
- Slideshare: slideshare.net/sap1ens

EXPERIENCE

Mobify

Senior Software Engineer, Lead, November 2016 - Present | Vancouver, BC, Canada

I joined Mobify to work on Mobify Platform, which includes Progressive Mobile and Engagement Marketing solutions. More details can be found here: https://www.mobify.com/platform/.

Bench Accounting

Director of Engineering, October 2015 - October 2016 (1 year) | Vancouver, BC, Canada

Engineering Lead, April 2014 - September 2015 (1 year 6 months) | Vancouver, BC, Canada

Software Engineer, September 2011 - March 2014 (2 years 7 months) | Omsk, Russia (Remote) -> Vancouver, BC, Canada

I'm the first employee of the company and I contributed a lot to its growth and success. In the last 5 years:

- Bench received $25M+ in funding
- Bench had grown from 6 to 250+ employees
- Engineering team had grown from 3 to 30 engineers

My contributions touched the whole stack. When I joined the company I mostly worked on a front-end, rewriting multiple single page web apps, then focused more on API, back-end of the core services and integrations, and finally I was

responsible for the whole Platform, including infrastructure work, DevOps tools and practices, core back-end services, integrations, microservices architecture and more.

I always cared a lot about projects we develop and products we ship. Everything from tiny UI elements to code quality and team motivation. I quickly became a leader for many initiatives and was promoted to the Engineering Lead. As an example, I led the transition from Java monolith app to Scala microservices.

My main projects and achievements included:

- Completely rewritten front-end app from tangled jQuery to CoffeeScript, Backbone.js and RequireJS (it was 2012 ;-)
- RESTful HTTP API for client and financial data [Scala/Jersey], as well as various APIs for other services and integrations [Scala/Spray]
- Billing/subscriptions functionality via integration with Stripe [Java]
- Highly-available J2EE/Tomcat setup with zero-downtime deployment
- Scala/Akka microservice template, supporting tooling and libraries. Now it's used for 10+ services
- Realtime eventing infrastructure that powers all notifications and chat messages [Scala/Akka/ActiveMQ/Camel/Postgres]
- Various integrations including Salesforce, Zuora, FileThis, etc. [Scala/Akka]
- Parsers/scrapers for multiple financial institutions [Scala]
- End-to-end testing for the whole system [Selenium/Java and Intern.js/CoffeeScript later]
- Creating reproducible infrastructure environments for the microservices architecture [Ansible, AWS CloudFormation]
- Highly-available and auto-scalable set of microservices with zero-downtime deployment, unified monitoring and logging [Docker, AWS Elastic Beanstalk, Ansible, Jenkins, Datadog, Splunk]

I continued to improve not only our codebase and quality standards, but lots of processes like project management, onboarding experience and education. In my role as a Director of Engineering I was responsible for managing multiple teams working on Platform, infrastructure, API and integrations, as well as participating in technical discussions, contributing to architecture and mentoring. Still, my sleeves were always rolled up and I was helping teams with regular tasks all the time.

PureBulk

Senior Web Developer, March 2010 - July 2011 (1 year 5 months) | Omsk, Russia (Remote)

I worked on a big e-commerce website (up to 20k+ active users daily), responsibilities included:

- Maintenance and feature development (full-stack), including catalog and shopping cart optimizations, custom admin UI and more
- Implementation of a special shipping module based on a packing optimization algorithm
- Heavy front-end development using various JS frameworks
- Zencart -> Magento migration
- Lots of front-end and back-end loading speed / performance optimizations

Aconcept

Web Developer, November 2009 - February 2010 (4 months) | Omsk, Russia

- Developed web sites and web apps (full-stack)
- Led training program for interns

IA Globus

Junior Web Developer, January 2007 - June 2007 (6 months) | Omsk, Russia

- Worked on a Russian music social network (full-stack)
- Implemented various web sites for companies in Russia and UK

EDUCATION

Omsk State Technical University, 2007 - 2012 | Omsk, Russia

Engineer's Degree, Software Engineering and System Analysis

Additional certifications and workshops

- Writing Microservices with Lagom, September 2016 | Strange Loop
- Deploying and scaling applications with Docker, September 2016 | Strange Loop
- Introduction to Rust, September 2016 | Strange Loop
- Implementing Domain-Driven Design workshop, March 2016 | by Vaughn Vernon
- Introduction to Machine Learning Using Spark, May 2015 | Polyglot Unconference
- Principles of Reactive Programming, May 2015 | Coursera
- Real World Programming With Haskell, May 2014 | Polyglot Unconference
- Functional Programming Principles in Scala, June 2014 | Coursera

SKILLS, PRACTICES AND TECH

I have experience in distributed systems, databases, message queues, caches, various APIs, integrations, SOAP, REST, HTTP, XML, JSON, JSON-Schema, RAML, Design Patterns, Actor systems, Event sourcing, CQRS, SOA and Enterprise Integration Patterns.

I'm a big advocate of the functional programming and Domain Driven Design. I work in Agile (Scrum and Kanban) project and product management, use TDD/BDD when needed and write tests extensively (Unit, Service, Integration, End-to-end ...).

I'm also really excited about microservices, containers and DevOps tools and practices (such as cloud infrastructure, configuration management, Continuous Integration/Deployment/Delivery, monitoring, logging, security, ...).

Tech I've used professionally over the last 10 years

- Scala: Akka (Actors, HTTP, Cluster, Persistence), Spray, Slick, SBT.
- Java: Spring Framework (Context, Security, Data, Roo, MVC), Hibernate, Tomcat, Jersey, Struts, Camel, Maven.
- JavaScript: ECMAScript3-6, CoffeeScript, Node.js, restify, npm.
- Python: Django, Flask.
- PHP: Zend Framework, Code Igniter, Zencart, Magento, Wordpress.
- Web: HTML/CSS, jQuery, Backbone.js, AngularJS, React.js.
- Testing: Selenium, JUnit, TestNG, ScalaTest, Jasmine, Intern.js.
- Misc: ActiveMQ, Postgres, MySQL, MongoDB, Redis, Ansible, Docker, Vagrant, Consul, nginx, Amazon Web Services (EC2, S3, RDS, ElastiCache, IAM, Lambda, CloudFormation, EB, ECS, ...), Git, SVN, Bash, NewRelic, Datadog, PagerDuty, Splunk, Jenkins, CircleCI.

Tech I've not used professionally, but either studied or used in small projects

Haskell, Erlang, Go, Kafka, Cassandra, Riak, Spark.

Thanks!