

# Alexandru Tetelea

Software Engineer

Email: alexandru.tetelea@gmail.com

Phone: +(40) 754 589 775

## Hi, I'm Alex, and I'm a Software Engineer.

I'm a highly skilled Scala and Java developer with experience in application development using Spring, Spark, Kubernetes and Kafka. The choice of **Java** and **Scala** - comes from the big community and a huge number of successful projects built with them.

I would like to work in intensive data processing projects, because "90% of all data which we have has been created in the last two years" and history shows us that companies which are adaptable and rely on data are more successful.

**Skills:** **Java** , **Scala** , **Kafka** , **Streaming processing** , **Spring** , **Kubernetes** , **Docker** , **Spark** , **Databricks** , **Fast learning**

## Experience

---

### E.ON

(Software Development)

#### Scala Software Engineer

Jun 2018 - present

- Dev:

**Scala**

**Apache Spark Batch/Streaming**

**Apache Flink**

**Databricks**

**Kafka**

- Ops:

**Kubernetes (Azure AKS)**

**Bash**

**Docker**

**GitLab CI/CD**

*"The more you know, the more you know you don't know."*

I'm part of the team which is providing POC for Big Data projects. My role in the team is to write all required logic which can transform data and expose them via **Kafka** or **API Services**.

To deploy spark jobs I'm creating **GitLab CI/CD** pipelines and all required components to access **Jobs Metrics** and expose logs via Logstash.

For one project in order to keep a service updated I had to write the logic which was extracting changed rows from DB snapshots in CDC way.

In this team I learned that, development is not just writing app code, it's also infrastructure setup, pipelines configuration and business logic decoding.

Possibility to learn things like Infrastructure as code, **Cloud Native** applications, **Streaming frameworks** and **Event driven** development allow me to find cleaner solutions to day by day tasks.

Moving from **Java** to **Scala** proved to be a mindset improvement, the functional approach of solving problems being more fluid and readable (until you start to use recursion).

# Endava

## Java Developer

Apr 2016 - Jun 2018

- Dev:

Java 8

Spring Cloud

Microservices

- Ops:

Jenkins

Grafana

- Learned

Clean Code

Code review

Agile team

Feature toggles

CI/CD

*And here started the first long term project. Here I learned that it's not enough to have a compiling code, it also need to be clean. Working more time on the same project I had moments to be ashamed by my own old code :D (never judge code before looking to git history ;))*

I was working for a non-banking financial institution. System was built as a big monolith with some Microservices around it, which were the result of a partially split of that monolith. Entire development and deployment respected Continuous integration and partially Continuous delivery and a suit of Unit/Integration/Component/ End to End test assured that every merge was keeping code standards high and business logic correct.

Applications were written with **Spring Cloud** and deployed using **Jenkins** on bare metal machines. Our team was the owner of Risk Assessment flow, providing data to a service which was taking the decision of accepting clients. **Code review** was a mandatory part of each merge request and presence of seniors implied many times refactoring of original code, without taking compromises.

From this project I learned that a number of well documented steps can make entire development process easier and less bug prone.

# Unicredit

(Business Integration Solutions)

## Junior Java Developer

2015 (Internship 3 Months)

- Dev:

Java

Servlets

*My first experience in production level software.*

At Unicredit I was part of the Maintenance team - which was fixing issues in the Online Banking Web Application. As a **Java Developer** I had to come with fixes to production issues, all of them implying coding.

The approach was a waterfall one with once a month deployment and code freezes for testing.

# Education

---

## Alexandru Ioan Cuza University (UAIC)

### BSc Computer Science

2013 - 2016

### MSc Computer Science

2016 - 2018

*The Alexandru Ioan Cuza University offers study programmes in Romanian, English, and French. In 2008, for the third year in a row, it was placed first in the national research ranking compiled on the basis of Shanghai criteria. In the 2012 QS World University Rankings, Alexandru Ioan Cuza University was included in the Top 700 universities of the world, on the position 601+, together with three other Romanian universities.*

#### Title: Bachelor of Computer Science

During my time at UAIC I learned that hard skills are not enough, many things can't be done alone and a team work is essential. I learned how computers works at low level (ASM programming) and TCP packets are sent over fiber. I learned many programming languages, starting with C and finishing with Haskell.

I keep thinking that at work you can learn many things but the core knowledge of a developer comes from university.

In these 3 years I was touching all aspects of computers science - Networking, Linux, Object oriented programming, Cryptography, Functional programming, Domain Specific Languages, Web Development, Data Structures and a 1 semester of crazy Math.

# Courses

---

## Bitdefender Training

2014 - 1 Month

During this training I learned WinApi, Reverse Engineering, Work with IDA Pro and Python

## CloudBase Training

2015 - 1 Month

This was an intensive, 8 hours training on Networking, Bash, Git and Python. The training finished with a practical test.

Hobbies :

Reading books

Experimenting with my Raspberry pi 2/3