CV - Marat Melgizin

Personal Information

Email: m.melgizin@yandex.ru
GitHub: https://github.com/m-melgizin
LinkedIn: https://www.linkedin.com/in/m-melgizin/
Telegram: @v_oxel

Professional Profile

I specialize in backend development with Kotlin and Java, focusing on building microservice architectures, monitoring systems, and data analysis utilities. I have hands-on experience in developing research and cryptographic solutions, as well as applying ML algorithms. I take an engineering approach, emphasizing clean code, reliable testing, and reproducible pipelines.

Work Experience

Java Backend Developer, T-Bank (mar. 2025 – p.t.)
Project – Microservice for Monitoring Performance Degradation of Modules and Methods
Tech stack – Kotlin, Java 17, Spring Boot, JUnit 5, Mockito, Testcontainers, GitLab CI/CD, Docker
- Developed a microservice using Kotlin, Spring Boot, and PostgreSQL
- Set up GitLab CI/CD pipelines
- Enforced code quality standards and best practices
- Covered business logic with unit and integration tests (JUnit, Mockito, Testcontainers)
- Participated in task planning and performed code reviews

Java Backend Developer, Consulting-Audit-Expert (sep. 2024 – mar. 2025)
Project - Internal Report Generation Service
Tech stack - Kotlin, Java 17, Spring Boot, PostgreSQL, JUnit 5, Mockito, Testcontainers, Docker
- Developed REST API for the report generation service using

Kotlin, Spring Boot, and PostgreSQL
- Wrote automated tests to improve code coverage and stability
- Configured development and testing environments using Docker
- Participated in task planning, conducted code reviews, and mentored team members

Java Backend Developer, AxxonSoft (oct. 2021 – sep. 2024)
Project – Backend for VMS
Tech stack – Java 8, PostgreSQL, JUnit5, Mockito, gRPC, Docker
- Developed server-side business logic for VMS using Java and PostgreSQL
- Tuned PostgreSQL indexes and optimized database queries
- Designed and implemented a new microservice for server-side event localization
- Developed and maintained gRPC APIs for microservice communication
- Wrote unit tests to ensure code reliability
- Participated in code reviews and onboarding of interns

Education

Bachelor's in Information Security, KBSU (2021–2025)
Specialization – Organization and Technology of Information Security
Thesis – Applying Machine Learning Algorithms to Detect Potentially Malicious Executable Files

Publications

Applying Machine Learning to Identify Phishing Websites Using a Decision Tree Classifier.
Akbasheva E.A., Akbasheva G.A., Melgizin M.E.
(eLIBRARY)

Machine Learning Methods for Detecting XSS Injections.
Ksenofontov A.S, Melgizin M.E., Shagirov A.S.
1st place at the "All-Russian Scientific and Practical Conference on Information Security"

Personal Projects

TheBreeder (Github)

A fast and practical tool for combining multiple .py files into a single script, simplifying distribution/deployment/encoding/encryption of Python scripts along with their dependencies
- Developed an algorithm to build a dependency graph of Python modules, analyze and resolve it
- Implemented an encoding algorithm for the final file to evade antivirus detection

CRINGE (Github)
CRINGE (Cryptographically Robust and Innovative Number Generator (Efficient)) – a cryptographically secure random number generator library based on the GOST R 34.12-2015 (Kuznyechik) block cipher
- Adapted CTR_DRBG RNG algorithm to work with GOST R 34.12-2015
- Implemented the algorithm as a library and utility for generating cryptographically secure random sequences

DominantTones (Github, Kaggle)
C++ library with Python bindings for image clustering and extraction of dominant tones
- Implemented image file reading
- Implemented KMeans clustering algorithm
- Developed Python bindings for C++ classes and methods

AI Code Detector (Github, Streamlit)
The main goal was to create a binary classifier that takes code as input and answers "Was this code generated by an LLM?"
- Researched existing solutions to this problem
- Developed a dataset collection script
- Participated in Streamlit app development

UDC Classifier (Github)
The main goal of the project was to develop a classifier that determines the UDC code of an article based on its abstract
- Researched existing solutions
- Helped develop a scraper for scientific articles
- Conducted experiments with various classifiers

Skills

Programming languages: Kotlin, Java, SQL, Python, C++
Databases: PostgreSQL, SQLite3

Libraries and frameworks: Spring, Hibernate, JUnit, Mockito, Testcontainers, gRPC
Development tools: Linux, Docker, Git, CI/CD, REST
Soft-skills: mentorship, teamwork, communication, code review, autonomy, documentation, knowledge sharing and training

About me

- Teaching Assistant at HSE University in courses: Python for Data Analysis, Applied Python, SQL, Mathematics for Data Analysis
- Mentor in VK Education programs