PAVEL**KOVALENKO**

Junior Java Software Developer

kovalenkopavelb@gmail.com | 89214332531 | kovpavel | Saint-Petersburg

EDUCATION

SPbSTU | *Bachelor of Software Engineering* **2020 - 2024**

- **GPA**: 4.7
- **What was studied**: DBMS, Microservice Architecture, Client-Server Applications, Programming of Microcontrollers, Networks and Telecommunications, System Design, Machine Learning & Neural Networks, Project Management
- **Stack**: Java, PostgreSQL, Spring (Boot, Web, Data JPA, Security), REST, Docker/Docker Compose, Python, C++, Matlab

Technopolis | *Java-developer of high-load applications* **2021 - 2023**

- **GPA**: 4.6
- **What was studied**: SQL/NoSQL, HighLoad, Application Profiling, Machine Learning, Mobile Development, Basic Frontend (HTML, CSS, JS/TS), Information Security & Blockchain, Testing
- **Stack**: Java, MS SQL, Apache Cassandra, Apache Spark, Apache Zeppelin, Apache Kafka, Apache Airflow, Hadoop, Scala, Solidity, Selenium/Selenide, Espresso/Kaspresso, Kotlin

EXPERIENCE (TOTAL - 1 YEAR 3 MONTHS)

Dzen | *Java Developer Intern* **March 2023 – July 2023 (4 months)**

Stack: Java, YQL, YDB, Yandex.Monitoring, Apphost, Python, Jira / Yandex.Tracker

I developed a news feed for dzen.ru. In more detail what did I do:

- I created a **monitoring and alert system** for the service to monitor its performance
- I conducted several **A/B experiments** that have shown a lack of demand for some functionality from users
- I added some **offheap fields** and sent them through several services working with **APIs** and changing them
- I implemented a **container of widgets** (weather, geoposition, rates of currencies, etc.) for simplifying the adding of new ones
- I also wrote **automatic Python scripts** to refactor a huge amount of similar code (6k+ lines)
- In addition to the routine task flow, I **created a documentation** about monitoring systems, entities and its communication in Dzen at all that my colleagues used
- Sometimes I **organized the entire project team** in Scrum StandUps

Technopolis | *Java Developer* **Feb 2023 – June 2023 (5 months)**

Stack: Java, Spring (Boot), PostgreSQL, TamTam API, robbi.ai

I developed a graduation project «RandomCoffee Bot» in the team according to the Odnoklassniki business case to increase social communication within the company. In more detail what did I do:

- I created **a PostgreSQL database class interface** to organize and access all application data
- I added the ability to **save the history of user connections** to improve the quality of the search for interlocutors
- I **refactored a code** to improve its readability

SPbSTU | *Python Developer* **Feb 2022 – June 2022 (5 months)**
Stack: Python (PIL), SQLite, Telegram API, Docker
I developed a Telegram bot in the team according to the Belorusskaya Cosmetika business case to protect photos from plagiarism in the Internet. In more detail what did I do:

- I created **a SQLite database class interfaces** to organize and access all application data
- I added **the processing of photo archives**
- I **prepared the documentation** for the project
- I **interacted with the customer** and looked for a common language to solve problems with him

SPbSTU-based factory | *Java Developer* **Oct 2021 – Jan 2022 (4 months)**
Stack: Java, Spring (Boot, Web, Data JPA, Security), HTML, CSS, JS, DeepSource
I developed a Tracking System in the team according to the SPbSTU-based factory business case to track the progress of work on the machines by the employees of the enterprise. In more detail what did I do:

- I implemented a **global timer** to see the current time in UI
- I implemented **some endpoints** of the system server
- I **interacted with the customer** and looked for a common language to solve problems with him

PROJECTS

Technopolis | *Java Developer* **Feb 2022 – Dec 2022**
Stack: Java, Cassandra, Lua, JUnit, one-nio, wrk2, async-profiler
I developed my own NoSQL DB by myself as part of the VK educational project, Cassandra was taken as a prototype. In more detail what features did I do:

- As a basic part of DB, I added **Persistency** to save data, **Range queries** with binary search and **Compaction** to remove old data
- As a part of network communication with DB, I added **HTTP REST API** to get data by endpoints, **Asynchronous server** to unload SelectorThreads, **Sharding** to up horizontal scalability, **Replication** to ensure fault tolerance, **Inner Asynchronous communication** within the cluster nodes, **Range queries** with HTTP Chunked Response
- Also I profiled my DB and tested its performance actively, and got detailed feedback from experts after each feature

SKILLS

Technical Java, Spring (Boot, Data JPA, Web, Security), PostgreSQL, Cassandra, Docker, Python, C++, Yandex.Monitoring, Jira

Language English - B1

READING LIST

Robert C. Martin «Clean Code»

CERTIFICATION & AWARDS

2024 EF SET English B1 Certificate

2023 Technopolis Graduation Certificate