

IGNAT GOMZYAKOV

+79523047827 ◇ Saint Petersburg, Russia
psajd83@gmail.com ◇ [Github](#)  ◇ [Telegram](#) 

EDUCATION

Bachelor of Software Engineering,
ITMO University

Saint Petersburg, Russia
2021 - 2025

SKILLS

Programming languages	Java & Kotlin, SQL, HTML, CSS, JS
Technologies	Spring framework, Ktor, Exposed, JUnit, Thymeleaf, JDBC, Hibernate
Developer tools	Git, Jenkins, Docker, Maven, Gradle
Languages	Russian(native), English (B2)

EXPERIENCE

Backend developer
[Lipt-soft](#)

June 2023 - August 2023
Saint-Petersburg, Russia

- Engaged in the transfer of a separate service to a new framework for testing using BDD. (JUnit to JBehave)
- Engaged in the development of the company's infrastructure projects: connecting Google OAuth, configuring the interaction of several internal services.
- Fixing bugs on the frontend side, interacting with the React framework

Freelancer
[MyLabs](#)

March 2022 - May 2023
Remote

- Engaged in the execution of programming orders: the implementation of business logic or algorithmic tasks.
- Faced with the use of a previously unknown technology stack.
- Completed more than 70 works to order.

PROJECTS

- **[Banks manager](#)** Banking application that allows users to manage banks, their customers, and customer accounts. Implemented a Spring MVC application. The frontend part is implemented using the Thymeleaf template engine and the Bootstrap library.
- **[Telegram Quizlet bot](#)** Telegram bot, created in the form of a compact replacement for quizlet cards, which help to quickly memorize the necessary information. The bot is configured using telegram web hooks and Spring Boot.
- **[Reports maker](#)** An application for compiling reports on the work of company employees. The application was designed using a three-layer architecture. The sqlite DBMS was used as a data warehouse. The presentation layer was implemented as a console application
- **[Backups](#)** An application for creating recovery points in virtual memory and on a local repository. In the process of doing the work, many design patterns were studied. It also created its own logging mechanism and used a library for serialization and deserialization of files to save the application state.