Kirill Degtyarev

Senior QA automation engineer (Java developer)

Phone: +375336617771

Address: Belarus, Minsk

Website: https://www.linkedin.com/in/kirill-degtyarev-655423116

Email: sensesnet@gmail.com

I have over than five years of experience as Software Quality Assurance and Automation testing. I skilled in different types of testing such as functional testing, WEB app testing, mobile app testing, GUI app testing, system regression, performance testing.

I used a lot of solutions from testing world, all of that I realized at own tests frameworks. My goal, it helps to save time for acceptance or full regression test runs, have result in a few hours and speed up the development process. I practiced of using continuous integration system (pipeline, nightly runs, triggers).

I'm active person, looking some powerful and new (courses, technologies (like Spring Framework, JSF, GWT, Salesforce), hardware).

I'm sociable, so I have many friends in IT sphere.

Some my work you can look on: https://github.com/sensesnet

EXPERTISE

Stack: Scala; Java SE; Java EE (Spring; Servlets+JSP).

Types of testing: Functional testing, Automated testing, Performance testing; API testing.

AQA Pattern: MVC, Page-object; Page-factory; et al.

Building: SBT; Maven, Apache Ant, Gradle.

Data: JDBC, Hibernate; (**DB**: MySQL; DB2; Postgres).

BDD: Cucumber(Gerkin).

Functional testing frameworks: JUnit; TestNg; Sikuli; Selenium; Selenide; Winium; ScalaTest.

Performance testing: Jmeter(+ inside test framefork with report system); Blazemeter.

Server: Apache Tomcat; WebSphere; Nginx.

API: Apache-HTTP-client; Swagger.

Data structure: XML/XSLT, JSON.

File transfer protocols: SFTP, JSCH, IMAP.

IDE and tools: Intellij IDEA, Eclipse, MS Visual Studio, Git.

Reporting: Alure; NET.Masterthought, ScalaReport.

Continuous integration: JetBrains Teamcity; Jenkins; Atlassian Bamboo.

VCS: Git (Bitbucket, GitLub, GitHub).

Operating systems: Windows, Linux Ubuntu 20.04.

Doc: Confluence, javaDoc.

Mobile app (Native): Appium (Android/IOS app).

Mobile emulator: Genymoution

Deploy app: Docker

EXPERIENCE & PROJECTS (latest)

Evolution Gaming (Minsk, Belarus)
Nov 2020 to Present

Senior Scala QA automation engineer
Project: Backoffice platform

   My activities was connected with main game reporting system inside backoffice platform. The testing process included next steps:
- functional test framework creation for UI browser testing (e2e tests)
- test implementation (BDD) and preparing test data (JSON)
- the process of configuring local test run on Docker
- the process of running test inside Jenkins pipeline CI/CD, configuration process (manifest; MR)
- test result (standart scala report system)

Agile: kanban.
QA Team: *1 Project Coordinator, 1 Automated QA, 2 QA*
BE Team: *3 developers*
FE Team: *3 developers*

Stack: Scala; Cats; ScalaTest; Selenium; BDD; docker; CI/CD(k8s; Jenkins); JDBC.
Testing: e2e tests & integration tests

---

iFUTURE (Minsk, Belarus)
Nov 2019 to 2020

JAVA BE Developer | QA automation mentor
Project: OROPAY – money transfer system (about: ***https://www.oropay.com/***).

   The payment system is any system used to settle financial transactions through the payments transfer, and includes the institutions, instruments, people, rules, procedures, standards, and technologies that make such an exchange possible.
   Developing functionality on backend side, work with service-oriented architecture. Analysis new features.

Agile: 2 weeks sprint.
QA Team: *1 Project Coordinator, 2 Automated QA, 3 QA*
BE Team:

Stack: Java 8; Spring Boot; Web Services (REST); JDBC template.
Testing: unit & integration tests

---

iFUTURE (Minsk, Belarus)
July 2018 to Nov 2019

Lead/Senior QA automation engineer

   I realized BDD acceptance and full regression testing, test cases coverage was created with QA and BI teams.
   During the project, I connected with dev. team and used all possibilities for testing such as (API (Swagger), SCH, SFTP, IMAP, DB, Logs).
   It helped me to create preconditions for test and create independent test from each other. I realized multithreading runs (used JUnit runners + Maven profiles) and integrated project at Bamboo.
   I used only high level test cases for BDD descriptions and it helped to have very readable and useful report. I added to report browsers screenshot after each of error. I had own building server for testing used Docker + Nginx.
   The main idea was created UI tests for three WEB payment clients.

Agile: 2 weeks sprint.
QA Team: *1 Project Coordinator, 1 Automated QA, 3 QA*

Stack: Java SE v.10; (Selenium WebDriver; API; JDBC; JSCH; SFTP; IMAP; Maven).
Pattern: Page-object; et al.
Data: JDBC, Postgres.
Test FW: JUnit + CUCUMBER.
File transfer protocols: SFTP, JSCH, IMAP.

Test server: Docker + Nginx.
CI: Bamboo.

Godel Technologies (Minsk, Belarus)

QA automation engineer

Project: *Earthport* – money transfer system (about: **https://www.earthport.com/about-us/**).

   The project was connected with payments to almost any bank account in the world, delivering significant cost and operational efficiencies. I realized BDD acceptance testing, test cases coverage was created by QA team.
   We tried using 'Shift Left Testing model' for test creation and this conception gave results. Sometimes I started to create test coverage before than feature was created by developers.
During the project, I connected with dev. team and used all possibilities for testing such as (API, SCH, SFTP, DB, Logs).
   It helped me to create preconditions for test and create independent test from each other. I had own test environment and used WebSphere as a building server.
 I used only high level test cases for BDD descriptions and it helped to have very readable and useful report. I added to report browsers screenshot after each of error. We integrated project at JetBrains Teamcity.
The main idea was created acceptance UI tests for WEB payment clients.

Agile: 2 week sprint.
QA Team: *1 Project Coordinator, 3 Automated QA, 6 QA*

Stack: Java SE v.8; (Selenium WebDriver; API; JDBC; JSCH; SFTP; IMAP; Maven).
Pattern: Page-object; et al.
Data: JDBC, DB2.
Test FW: JUnit + CUCUMBER.
File transfer protocols: SFTP, JSCH, IMAP.
Test server: WebSphere.
CI: Teamcity.

ALFA-BANK (IBA) (Minsk, Belarus)

QA automation engineer

Projects: **Alfa-click**(WEB); **Cashbox**(GUI); **InSync**(native) – WEB, UI and mobile client-bank applications, about (**https://www.alfabank.by/**).

Alfa-click (WEB Client-Bank**)**

   The system "Client-Bank" allows creating a flexible and mobile interaction system between customer and the bank via organization of integrated electronic service.
The main idea of tests project was create full regression control and describe full process flow for new comers. We tried to use BDD testing because it helped to describe all process flow and connected with UI tests. Only high level steps were used for tests coverage.
During the tests project we created tests model, described all process flow and realised project. Interesting moments for me was using Page Factory as a main test framework and Java Reflection for locator implementing.

Stack: JavaSE; JDBC (MySQL); Selenium WebDriver;
JUnit + Cucumber API; Java Reflection API; Regular expressions (main pattern PageFactory).
QA Team: *1 Project Coordinator, 1 Automated QA, 3 QA*

Cashbox (**GUI** app)

Cashbox is a relatively simple application for Windows that manages finances. It allows the bank cashier to manage multiple accounts with many convenient features. Cashbox is currently under active development and many features will be added in the future.
   Application was created on java 6 with Swing API. It looked like a challenge by next reason:
      - Products Company did not provide the source code;
      - I had empty window when tried to find locators, all object at app.window was invisible for 'AutoIt' and other app.
   I tried used 'Sikuli' framework for auto tests building. Before that I find other way for automation and investigated new frameworks like 'Winium'. 'Winium ' couldn't find any locators too but easily found active frame on windows desktop. I decided to use both frameworks.
   Application was installed on remote win server, 'Winium' helped to start start/stop application and found active app. windows

space. 'Sikuli' used graphics object and automated operations at active space.
BDD was added to project for Bank support team, this helped to learn full transaction flow for new comers.
Each new application release is testing on bank side by support engineers.
This project gives info about bugs on main transaction flow and protects all bank cashbox from idle.)

Stack: JavaSE, Winium, Sikuli, Regular Expression, Cucumber, JUnit (main pattern Page Objects).

QA Team: *1 Project Coordinator, 1 Automated QA, 3 QA*

InSync (native app.)

InSync is - reliable, technological and convenient mobile banking.
I decided to use 'Appium' automation framework for use with native. Only android native was covered and tested during the project. I covered performance tests by JMeter.
The project included test coverage for main fifteenth operations.

Stack: Java SE; (Selenium WebDriver; JDBC; Java Reflection, Maven, TestNg; + CUCUMBER).

Performance: Jmeter

Emulator: Genymoution

QA Team: *1 Project Coordinator, 1 Automated QA, 3 QA*

EDUCATION & COURSES

Belarusian State University of Informatics and Radioelectronics
2005 to 2010

bachelor degree

Study Faculty is 'Information Technology and Management in technical systems'.

Activities and Societies: http://www.bsuir.by/en/about-bsuir

Participate in 25 scientific conferences .

Have three scientific innovations in electronics .

Belarusian State University of Informatics and Radioelectronics
2010 to 2011

master degree

course "Automation of engineering systems"

Automation engineer

Activities and Societies: http://www.bsuir.by/en/master-degree

Create project about automation process from gas storage park.

IBA Minsk (Java courses)
2013 to 2014

practice

(OOP; Identifiers; Expressions and Flow Control; Class Design; Exceptions&Assertions;
Collections and Generics Framework;I/O; Swing API; Threads; Networking; ANT; Pattern)

IT Academy (Java EE courses)
2015 to 2016

practice

Activities and Societies: https://github.com/sensesnet

JSP + JDBC (MVC - architecture, Maven, Tomcat, Git)

JSP + Hibernate (MVC - architecture, Maven, mapping, annotation, HQL, transaction)

Spring+Hibernate(MVC - architecture, Maven, Beans, SpEl, Spring MVC, Spring Security, REST)

IBA Minsk (Java courses)
2016 to 2017

practice

- Selenium WebDriver (locators, waits);

- Data-Driven testing;

- Behavior Driven Development

(Cucumber JVM + Selenium Webdriver, JBehave + Selenium Web driver;

- Page Object/ Pagefactory;

- PhantomJS and SlimerJS;

- Selenium GRID

IT Academy (JS courses)

practice
2018 to 2019

  HTML CSS; JavaScript; JQuery
(DOM, SVG, Render tree, MVC, AJAX, SPA).

IT Academy (JD2 courses)

Java 8; (JDBC; JSP; Servlets)
(Hibernate; Spring 5; Rest; Spring AOP)

*2019
(Sept-Nov)*

LANGUAGES

English (Intermediate level)