

Daniel Galvis Torres

dangaltor@gmail.com

Phone number: (+57) 3057992327

<https://github.com/danielgalvis98>

<https://www.linkedin.com/in/daniel-galvis-torres/>

Senior Software Development Engineer / Java

SUMMARY

Software Development Engineer with over four years of experience implementing Spring Boot and .Net applications, and one year contributing to React application, building microservices with shift Left Testing Strategies. Good teamwork skills to materialize the testing pyramid and the technical skills required to develop the frameworks to implement those tests. Also, experience configuring and maintaining CI/CD pipelines to verify the quality of the software through a Continuous Testing strategy, including stages such as Build, Linting, Test, Static Code Analysis (Sonar), Automatic Semantic Versioning, and pushing Docker images to registries. Openness skill to listen and discuss different ideas and strategies to achieve the team's goals.

Solutions

- Spring Boot Applications
- SPA with React
- Azure Storage Services with .Net and Spring Cloud Azure
- CI/CD Pipelines with Jenkins and Azure Pipelines
- API E2E Tests with Rest Assured
- Component/Integration Tests on .Net, Java
- Component/Integration Tests on React with Testing Library
- Contract Tests with Pact
- Integration Tests against Databases and Azure Storage Services on .Net using Docker
- Project and Sprint management with Jira and Azure DevOps

Technologies

- Spring Boot
- Spring Data JPA
- Spring Web
- Spring Cloud
- Spring Security
- Junit
- .Net
- Shift Left Testing
- Continuous Testing
- Sonar
- MySQL
- Azure Storage Services
- Jira
- Jenkins
- Azure DevOps
- C#
- Javascript
- Kafka
- React
- Testing Library

Professional and Business Experience

Software Development Engineer

December 2020 - Present

Perficient, Inc.

- Implement event-driven microservices to receive files uploaded by physical devices with TUS protocol, and process and manage those files. Services were mainly built with Java using Spring Boot, implementing Spring Kafka to consume and produce events, Spring Web to build the API endpoints Spring Data to connect to MySQL database, and Spring Cloud Azure to interact with Azure Storage Services. Some of the services were implemented with .Net too.
- Implement REST API microservices with Java using Spring Boot to manage data and loans from clients of a real state company. Also, Spring Web and Data were used, in addition to some configurations on Spring Security.

- Contribute to the development of React Application to manage data and loans from clients of real state company, while setting up Testing Library to add component tests with recommended practices, using in-memory Redux.
- Responsible for proposing and implementing an Automation strategy on Java and for the Java, Javascript, and .Net projects using shift-left testing, strongly focusing on lower levels of tests based on the test pyramid, allowing for reduced feedback time cycles, preventing bugs before code is merged, reducing the total of required end-to-end tests and speeding up release cycles.
- Worked with team to find ways to refactor (With good practices) code to make it more testable. For component tests, worked on building and maintaining .Net libraries to create Kafka Fake and Azure Storage Queue fakes to facilitate implementation of those tests. For integration tests, I worked with Azurite on Docker to test integration with Azure Storage Services.
- Socialize strategy with the team to implement lower-level tests, build the test setups, add candidate tests, and discuss best practices, allowing dev, QA, and DevOps to be aligned.
- Work with DevOps to build CI pipelines on Jenkins and Azure Pipelines that run on every PR and every merge to check the build, run automated tests, static code analysis, automatic versioning, build, tag, and push Docker image to the docker registry.
- Configure JTE (Jenkins Template Engine) on multiple projects, to have a CI parameterizable template for all projects, forcing the implementation of important steps on all projects and allowing the creation of new pipelines faster.
- Configure SonarCloud for every project, customizing Quality Profiles and Quality gates, to analyze every PR and block it if the minimal quality gate metrics were not accomplished.