Baptiste Calin

Freelancer

baptiste.calin@gmail.com +32 478 94 10 65 Colfontaine, BE linkedin.com/in/baptistecalin

baptistecalin.github.io

Software Engineer

With more than 15 years of experience in software engineering, I bring strong expertise in application development, architecture, and quality testing. My background spans full-stack development (Java, Node.js, React, Spring Boot) and CI/CD automation, allowing me to deliver robust, scalable, and high-performing applications.

I specialize in bridging software engineering best practices with modern DevSecOps, test automation, and agile delivery. My experience includes building enterprise applications, leading development teams, and implementing continuous integration pipelines to improve efficiency and product quality.

- Software Development & Architecture: Java, Node.js, React, Spring Boot, SQL, CI/CD
- Quality & Testing: Test automation (JUnit, Selenium, Jest, Karma), DevSecOps, performance & security testing
- Agile Delivery: Scrum, ITIL 4 practices, change and configuration management
- Tools & Platforms: GitLab, Jenkins, Docker, SonarQube, PowerBI, AWS

My goal is to design and deliver innovative, secure, and maintainable solutions while ensuring high quality through continuous improvement and testing.

Soft Skills

- Leadership / Ownership
- Team coordinator / leader
- Trainer & Coach
- Proactive
- Result oriented
- Innovation thinking
- Positive and invested

Profile skills

Processes	Agile / Scrum methodology, Configuration Management, ITIL 4 practices, BPMN
Management	KPI, Personal Development Plan, Recruitment
Development	Java/J2EE, Node.js, Pipeline, HCL, Python, HTML/JS/CSS, XML/XSLT
Frameworks	Jest / Karma, RESTful APIs, JUnit / TestNG, Angular, Selenium, Spring, Hibernate / JDBC, DHTMLX, JQuery
Tools	Jira, Confluence, GitLab, Jenkins, Github, SonarQube, Docker, VS Code, Microsoft Office Suite, Postman, Snyk, PowerBI, IntelliJ, Eclipse
Platform	Ainsible, AWS, CI/CD, JBoss, Tomcat, Linux
Databases	PostgreSQL, SQL Server, Oracle, H2

Work Experience

Quality Expert

SPW - Freelance | Apr 2023 - Sept 2025 | Namur (2 yrs 6 mos)

Application Lifecycle Management (ALM) with ISO 12207 - Analysis of the standard and identification of actions to be implemented - Project: Participation in setting up a development factory and CI/CD chain implementation at the SPW - SonarQube: Implementation of the tool in conjunction with the MoE project's DevSecOps team and developers teams

Skills used:

- **Processes:** Agile / Scrum methodology, Configuration Management
- **Management:** KPI
- **Development:** Java/J2EE, Node.js, Pipeline, HCL, Python
- **Frameworks:** Jest / Karma, RESTful APIs
- **Tools:** Jira, Confluence, GitLab, Jenkins, Github, SonarQube, Docker, VS Code, Microsoft Office Suite, Postman, Snyk
- **Platform:** Ainsible, AWS, CI/CD

Quality Expert

ZORGI | Jun 2019 - Apr 2023 | Heverlee (3 yrs 11 mos)

Medical Device Project management - Quality Management System implementation under Medical Device Directive (93/42/EEC) and Medical Device Regulatory (EU) 2017/745 - SaMD: Technical files consolidation - Certification ISO 13485 - Training Management - Records management - Authority and Notify body communication - ISO 13485 Quality Management System connection with ISO9001 and ISO27001 - Quality Management System connection with ITIL 4 practices - GDPR Project management + GDPR connection with ISO27001

Skills used:

- **Processes:** ITIL 4 practices, Agile / Scrum methodology, Configuration Management
- **Management:** KPI, Personal Development Plan, Recruitment
- **Tools:** Jira, Confluence, PowerBI, Microsoft Office Suite
- **Platform:** CI/CD

Test & Release Manager

Xperthis group | Jun 2020 - Jun 2021 | Nivelles (1 yr 1 mo)

Release software management Deployment planning - Test automation strategy and implementation - Quality Software management (Quality Assurance, Control, Managemnet) - Downtime management of IT System - Team management

Skills used:

- **Processes:** ITIL 4 practices, Agile / Scrum methodology, Configuration Management
- **Management:** KPI, Personal Development Plan, Recruitment
- **Development:** Java/J2EE
- **Frameworks:** JUnit / TestNG, Angular, Selenium, RESTful APIs
- **Tools:** Jira, Confluence, Jenkins, Microsoft Office Suite, Postman
- **Platform:** CI/CD

Solution Architect - Software Engineer

Xperthis group | Mar 2006 - Jun 2019 | Nivelles (13 yrs 4 mos)

Product management & Architect for a web-based healthcare application: Electronic Patient Record (EPR) - Saas Solution for Portail Patient Application - Set of teams coordinator and Team management - Incident consultant and helpdesk management

Skills used:

- **Processes:** BPMN, Configuration Management
- **Management:** KPI, Personal Development Plan, Recruitment
- **Development:** Java/J2EE, HTML/JS/CSS, XML/XSLT
- **Frameworks:** Spring, Hibernate / JDBC, DHTMLX, JQuery, RESTful APIs
- **Databases:** PostgreSQL, SQL Server, Oracle, H2
- **Tools:** Jira, Confluence, Jenkins, IntelliJ, Eclipse, Microsoft Office Suite
- **Platform:** JBoss, Tomcat, Linux, CI/CD

Education & Certifications

- AWS Cloud Practitioner essentials (2023)
- Certified ScrumMaster - CSM (2023)
- Person Responsible for Regulatory Compliance (2021)
- Coach and Manage a team (2021)
- ITIL 4 Foundation (2020)
- ISO 13485/2016 and CE Mark (2019)
- PostgreSQL DBA (2015)
- Bachelor’s in Informatics – H.E.P.M.B.C (2005)

Communication skills

- English : Limited working proficiency
- French : Native

The code below is used in many places, what would be the best way to improve it to make it reusable?

```
if (truc == Status.TO_DO) {
// do something
} else if (truc == Status.TO_DO) {
// do something
} else if (truc == Status.PROCESSING) {
// do something
} else if (truc == Status.VALIDATION) {
// do something
} else if (truc == Status.DONE) {
// do something
}
```

I'll use polymorphism in the Status enum to encapsulate the behaviour of each state. So, instead of having scattered if/else statements, each status implements its own method. This is more maintainable, readable, and changeable without breaking the entire code.

Class: TaskService.java

```java
public class TaskService{
    public void createTask() {
        System.out.println("Create task.");
    }
    public void processTask() {
        System.out.println("Process task.");
    }
    public void validateTask() {
        System.out.println("Validate task.");
    }
    public void archiveTask() {
        System.out.println("Archive task.");
    }
}
```

Enum: Status.java

```java
public enum Status {
    TO_DO {
        public void handle(TaskService service){
            service.createTask();
        }
    },
    PROCESSING {
        public void handle(TaskService service){
            service.processTask();
        }
    },
    VALIDATION {
        public void handle(TaskService service){
            service.validateTask();
        }
    },
    DONE {
        public void handle(TaskService service){
            service.archiveTask();
        }
    };

    public abstract void handle(TaskService service);
}
```

Class: Main.java

```java
public class Main {

    /*
    if (truc == Status.TO_DO) {
        // do something
    } else if (truc == Status.TO_DO) {
        // do something
    } else if (truc == Status.PROCESSING) {
        // do something
    } else if (truc == Status.VALIDATION) {
        // do something
    } else if (truc == Status.DONE) {
        // do something
    }
    */

    public static void main(String[] args) {
        TaskService service = new TaskService();

        Status statusToDo = Status.TO_DO;
        statusToDo.handle(service);

        Status statusProcessing = Status.PROCESSING;
        statusProcessing.handle(service);

        Status statusValidation = Status.VALIDATION;
        statusValidation.handle(service);

        Status statusDone = Status.DONE;
        statusDone.handle(service);
    }
}
```

Result:

```
Connected to the target VM, address: '127.0.0.1:62851', transport: 'socket'
Create task.
Process task.
Validate task.
Archive task.
Disconnected from the target VM, address: '127.0.0.1:62851', transport: 'socket'

Process finished with exit code 0
```