

Jean-Baptiste Lasselle

Cloud Engineer/SRE

Who I am

Je code avec **terraform, packer, pulumi, ansible**. Je plonge dans le code avec les devs, J'**automatise toute l'infrastructure**. J'implémente des terraform providers, des modules terraform, des collections ansible, pour normaliser le code écrit par les SRE / Devops. Je suis un évangéliste **Gitops** et **Infrastructure As Code**. J'aime comprendre, apprendre, chercher et trouver, implémenter de nouvelles idées, résoudre des problèmes avec mon équipe. J'aime les challenges.

My best friends list

Kubernetes, Terraform, Packer, Pulumi, FluxCD/ArgoCD, Atlantis, Ansible Tower, Circle CI, Gitlab CI, Tekton, Jenkins, Openstack, Ansible Tower (AWX), HashiCorp Vault/OpenBAO, Docker, K3D, Minio, Nginx, Traefik, Linux shell, TypeScript/NodeJS, React, Golang, Python, <https://gohugo.io> <https://astro.build>

I usually work with

Terraform, Packer, Pulumi, FluxCD/ArgoCD, Atlantis, Circle CI, Gitlab CI, Tekton, Jenkins, Ansible Tower (AWX), shell scripting (sh/bash)
Azure, AWS, GCP, OVH, VMWare vSphere, OpenStack

À l'Université, j'ai étudié les mathématiques pures.

J'ai commencé dans l'IT comme développeur.

2015, je me suis focalisé sur l'infrastructure (as code), et le monde du cloud.

J'adore coder, et tout automatiser.

Je suis [Certifié HashiCorp Terraform](#).

Aujourd'hui, je suis un **Cloud Engineer/SRE**, un évangéliste amoureux de l'automatisation. Construire des systèmes autonomes me passionne (Kubernetes Controllers, Operators, AutoScalability, self-healing abilities).

Mes Centres d'intérêts:

- Concevoir et implémenter des infras en **Infra-As-Code** (souvent avec approche gitops)
- Développer des **terraform providers**, des outils **OpenBAO**
- Développer des frameworks de modules Terraform/collections Ansible, pour normaliser le code des SRE/Devops. Les Landing zones: Quels frameworks / pattern utiliser, après le Cloud Adoption Framework?
- Développer des **Opérateurs Kubernetes**

Expérience Professionnelle

> Michelin

SRE Devops, Infra As Code SME

02/2024 - 01/10/2025

Project Context: équipe de 30 ingénieurs sur 5 continents. Michelin a une centaine de salle mini datacenters répartis dans le monde. Le projet vise à unifier leur exploitation, et à construire une offre infrastructure as a service (IAAS) pour leur consommateurs internes.

Tech stack: VMware vSphere, VMware Aria Automation, Ansible/Ansible Tower, Kubernetes, Docker, Packer, Hashicorp Vault, OpenTOFU, python, Artifactory, Gitlab pipelines

J'avais le rôle "Automation SME" dans l'équipe.

Achievements I am so proud of:

- J'ai introduit l'utilisation de Hashicorp `Packer`, le concept d'infrastructure immuable, de golden images, et de hardening des images dans les build packer.
- J'ai entièrement construit un service Ansible Tower, dans un cluster kubernetes cluster, avec automatisation complète des opérations d'exploitation du service Ansible Tower: provisioning, backup/restore. Le service Ansible Tower est géré avec l'Opérateur Kubernetes AWX <https://github.com/ansible/awx-operator> . Env. full Airgapped (isolation réseau complète), j'ai aussi automatisé la provision et les upgrades du cluster kubernetes K3S. Artifactory Private Image registry.
- J'ai entièrement automatisé la provision de VMs Windows server en à partir des golden images buildées avec packer, fichiers de réponses autounattend xml, et ansible pour exécuter les opérations de « post install ».
- J'ai conçu, implémenté le cycle de vie complet des environnements d'exécution Ansible Tower (AWX) avec Ansible Builder, et des Collections Ansible dans un contexte full airgapped - artifactory étant configuré comme repository ansible galaxy privé, idem pour les packages python (python wheels).

Duties:

- Je livre des POC (Proof of concepts), bench-marking des outils, je construis les environnements dans lesquels les outils d'automatisation s'exécutent
- J'évangélise les nouveaux concepts / nouveaux outils, les bonnes pratiques
- Je forme les membres de l'équipe à l'utilisation des nouveaux outils
- Je fais les revues de Merge Requests

> Limagrain

SRE Devops, terraform SME

09/2023 - 02/2024

Project Context: Chez Limagrain, une application critique pour le métier, disposait d'un seul environnement de dev, construit manuellement. J'ai été chargé de construire tous les environnements et en premier l'environnement de production qui manquait cruellement.

Tech stack: Terraform, Helm, kubectI, Docker, Azure AKS / App Gateway Ingress Controller (AGIC), Azure Functions, Azure CosmoDB, Azure EventHub, Azure Devops (Pipelines), Azure Managed identities / Service Principals, Azure networking - vnet, subnets, private endpoints, Artifactory, shell scripting (sh, bash).

On m'a d'abord demandé de mener un audit technique global du projet, sur le périmètre infrastructure, et CI/CD, avec comme livrable une note de synthèse/executive summary.

J'en ai tiré deux recommandations principales:

- il faut "tout automatiser": automatiser toute la construction de l'infrastructure, et il faut un pipeline qui automatise build, tests, deploy, à partir du code source.
- Il faut que les environnements soit isolés les uns des autres, dans lesquels seront déployées les releases, d'abord dans l'env de dev, puis staging, puis UAT (User Acceptance Tests), et enfin l'env de production.

Duties and achievements:

- J'étais en charge d'**implémenter** mes recommandations, en collaboration avec les 5 Développeurs, et l'équipe "Core Infrastructure" qui gère l'infrastructure au niveau global monde chez Limagrain.
- J'ai été invité aux "**architecture labs**", pour y faire une présentation de terraform et des concepts autour de l'Infra-As-Code. Ma présentation a été appréciée, si bien que d'autres équipes projet ont demandé à faire des workshops terraform avec moi.
- Je devais participer aux réunions de "validation gates": des audits qui permettent d'autoriser un projet à être déployé en production. J'y répondais aux questions des auditeurs quant aux choix de conception de l'infrastructure et de son automatisation.
- Avant mon arrivée, terraform n'était pas du tout utilisé, à mon départ, tous les environnements étaient automatiquement construit avec terraform: j'ai **convaincu les managers** de faire ce choix d'outil.
- En moins de 4 mois, j'ai introduit terraform et automatisé la construction de 3 env: DEV, UAT et production. Cela a permis d'identifier et résoudre deux problèmes complexes d'intégration entre composants cloud `Azure`, et Microsoft Dynamics.
- Les succès avec terraform, et mon travail d'évangélisation, associés au soutien des project managers, ont résultés en une véritable victoire: En Janvier 2024, l'équipe projet s'est vue accordée plusieurs souscription Azure dédiées, là où à mon arrivée une seule et même souscription Azure était utilisée et partagée par toutes les équipes projet de groupe.

> Michelin

SRE Devops Lead

11/2022 - 09/2023

Project Context: Michelin, Data Engineering / AI Project

Tech stack: Terraform, Kubernetes, Helm, Docker, Azure, Databricks, dbx, Snowflake, dbt, azcopy, Gitlab, Artifactory, shell scripting, Python, TypeScript, Astro, Hugo.

J'y étais en charge de la gestion de toute l'infrastructure Azure avec **terraform** (Azure Storage Accounts, vnet/sbnets, VMs, managed identities, service principals, etc...)

Ré-implementer, avec 2 Data engineers, la plateforme de data engineering. Dans cette refonte, j'ai travaillé sur les CI/CD pipelines du code python, en plus de la terraformation des ressources Azure.

En un mot, une **migration depuis** un stack **{Azure Data Factory / ADF Pipelines}**, vers un stack bien plus avancé, basé sur du code Python/PySpark, **{Databricks + Snowflake}**.

La nouvelle plateforme de data engineering traitait les données avec du code python: d'où la CI/CD Python.

Duties and achievements:

- Gestion des ressources Azure avec Terraform: Databricks workspaces, Azure Account Storage, networks, Azure Key Vaults, Managed Identities, R-Studio VMs, MongoDB clusters, etc...
- modularization du code terraform existant: modules terraform, terraform registry privé, terraform state stocké dans un Azure Account storage container.
- Conception des **CI/CD Modules** (Gitlab pipelines templates), des runners de type docker
- Conception, **Git workflows**, **Gitlab** release automation with gitlab-cli, python packages deployment to databricks (**databricks scheduled jobs**), snowflake (with liquibase).
- Construction de la Documentation technique de type Gitlab Pages, avec [astro](#) / [hugo](#)

> Gravitee Source

SRE Devops Leader

07/2020 - 03/2022

Project Context: Gravitee est une API Gateway Open Source

Tech stack: Pulumi, Terraform, Kubernetes, Helm, Docker, K3D, Azure AKS, AWS EKS, FluxCD. NodeJS, TypeScript, Python, shell scripting, Golang.

J'ai entièrement ré-implémenté la CI/CD de Gravitee, en la migrant de Jenkins, à Circle CI. JE peux aujourd'hui dire, que je suis un expert Circle CI

Je suis très fier d'avoir eu l'opportunité de travailler avec l'équipe <https://gravitee.io>, faite d'ingénieurs de très haut niveau, avec lesquels j'ai énormément appris sur de nombreux sujets.

J'y ai notamment beaucoup appris sur ce que sont de vrais systèmes cybernétiques dans le vrai monde de l'industrie du cloud: c'est par exemple là, que j'ai commencé à travailler sur les opérateurs Kubernetes.

Duties and achievements:

- Concevoir, implémenter, et migrer la totalité de la CI/CD Gravitee. J'ai développé un composant unique en son genre, avec NodeJS/Typescript et RxJS, permettant d'orchestrer l'exécution de dizaines de pipelines dans des git repository distincts. Ce qui a permis de se débarrasser des Single Point of Failures de la CI/CD précédente, qui tournait dans Jenkins.
- Automatisation des opérations standard d'exploitation de l'API Gravitee pour les offres "hosted" des clients (déploiement, backup, restore, upgrades, downgrades)
- Conception de l'offre SAAS de Gravitee API Gateway,
- Workshops clients pour les bonnes pratiques architecturales des déploiements des Gravitee API Gateway .
- Formation / évangelisation à l'utilisation de CircleCI des Dev Engineers
- Training junior devops engineers
- Level 3 Support pour les clients Gravitee, par exemple : mise en œuvre d'une VPN Gateway entre Azure et Google GCP

CRESH.EU

Cloud Engineer

04/2020 – 07/2020

Project Context: Cresh est une offre SAAS, dans le domaine des solutions de paiement en ligne (un peu comme snipcart).

Tech stack: Pulumi, Helm, Docker, Kubernetes, AWS EKS, Gravitee API Gateway. NodeJS, TypeScript, bash / sh shell.

Chez <https://cresh.eu>, une petite startup française, j'ai travaillé à la conception de toute l'infrastructure cloud, 100 % basée sur Kubernetes, sur AWS. AWS EKS, Helm deployments, `Gravitee` API Gateway. Les fondateurs de <https://gravitee.io> ont entendu parler de moi en raison du travail que j'ai fait chez Cresh, pour y mettre en oeuvre l'API Gateway Gravitee.

J'y étais le seul Cloud/Devops Engineer, j'y ai massivement utilisé Pulumi, Packer et Terraform.

En raison de points de divergence techniques majeurs avec le CTO, nous avons décidé d'un commun accord de mettre fin à notre collaboration.

Duties and achievements:

- Cloud Engineer, conception complète de l'infrastructure du projet
- Kubernetes SME

> SFR

Devops Engineer

07/2019 – 10/2019

Project Context: Déploiement automatisé des Edge datacenter de l'opérateur SFR - 5G.

Tech stack: Ansible, Docker, Gitlab EE, Jenkins Pipelines, LVM, Cisco NSO, Cisco KUBAM for PXEless boot, SSH Bastions, Python.

En tant que freelance chez SFR, je travaillais dans l'équipe Cisco NSO (SDN).

Duties and achievements:

- J'ai conçu et déployé un stack devops complet, pour que les développeurs NSO poussent sur git leur code python, et que celui-ci soit packagé, testé, et déployé sur les devices Cisco NSO. J'ai même été amené à automatiser un "build from source" de docker, en raison de contraintes de sécurité imposée chez SFR.
- J'ai évangélisé les concepts/bonnes pratiques de CI/CD auprès des devs python / NSO, avec une base Gitlab: ces développeurs étaient des ingénieurs réseaux, mais sans expérience de développement.
- J'ai conçu les git workflow, ai formé à l'administration d'un service Gitlab EE en tant qu'infrastructure: les ingénieurs en charge de l'administration du service Gitlab étaient des ingénieurs de développement sans expérience en matière de gestion d'infrastructure/Infra As Code.
- J'ai suivi une formation Cisco NSO pour l'automatisation en mode "bare metal provisioning" des clusters Cisco NSO, à Issy-les-Moulineaux, dans les locaux Cisco.
- J'ai travaillé avec une centaine d'ingénieurs appartenant à 3 équipes différentes:

- équipe Infrastructure pure (les “owners” de l’OpenStack),
- software development teams,
- network engineers teams

> Bosstek

Devops Engineer

04/2018 – 11/2018

Project Context: Infrastructure Management Consulting services.

Tech stack: Terraform, Ansible, Jenkins Pipeline, Docker (docker-compose) , NGINX, Gravitee.io (API Gateway), Keycloak, Kubernetes, ELK.

Salarié chez at Bosstek, une société de consultant en gestion des infrastructures, où j’ai vécu ma première expérience dans un contexte où je ne travaillais qu’avec des experts de l’infra, de le “pure infra”.

J’ai pris part à 3 projets, pour 3 clients différents de Bosstek. Cette expérience a été très enrichissante, j’y ai travaillé avec certains ingénieurs n’ayant jamais fait de développement, avec une longue expérience de gestion d’infrastructure “avec les anciennes méthodes”, notamment sans “infra as code”.

Duties and achievements:

- J’ai travaillé comme Cloud Engineer, au coeur des équipes projets infrastructure de très grosses sociétés: [SOPRA Steria](#), [ENEDIS](#), et [Carrefour One Cloud](#)
- J’ai été Team leader d’une équipe chez SOPRA STERIA, où j’ai travaillé sous l’autorité de David Maurange.
- J’ai mené à bien 2 études internes en tant que seul ayant une expérience Kubernetes:
 - Un POC bonnes pratiques Kubernetes Cluster Ingress Controller en Production.
 - Une proposition de solution pour un système de monitoring basé sur Rsyslog/Elastic Stack, (pour des serveur d’autorités de certification) : Le but était que la société puisse être certifiée pour la délivrance de certificats SSL/TLS.
- Au sein du projet Carrefour One Cloud (Un projet en collaboration avec Google), J’ai découvert le concept de Headless CMS, avec <https://strapi.io> , et c’est aussi là que j’ai pour la première fois travaillé sur un déploiement en production de l’API Gateway Gravitee.

> Vierling Communication

Java Dev Engineer – CI/CD SME

02/2016 – 01/2018

Project Context: R&D détection de pannes sur les lignes de fibre optiques dans des réseaux à grande échelle (OTDUs).

Tech stack:

- **Devops:** Ansible, PXE boot, PXE boot kickstart installations, Docker, Virtual Box, KVM, Free IPA Server, SSH / SCP/WinSCP, with a lot of bin/bashing.
- **CI stack:** Gitlab, Artifactory, MAVEN3 Git, Eclipse JBoss Tools, Jenkins, Checkstyle, JMeter, JUnit, DbUnit, Jasmine + Karma, Wireshark & wireshark dissectors
- **Dev stack:** Java, JCA IronJacamar, JPA, Hibernate, WILDFLY, EJB3 Stateless & MDBs', JAX-RS & Resteasy, Apache Shiro, Leaflet, Angular, Linux Pluggable Authentication Mechanism, mariadb, postgresql/postgis, .Net Windows Service Application component, Visual Studio.

Ingénieur de Développement Java, chez Vierling-Communication, une société de R&D du groupe « Laudren », il s'agissait de développer une solution de monitoring / détection de pannes sur des réseaux de fibre optique.

Ce fut une expérience très intéressante, dans la mesure où il ne s'agissait pas de détecter d'anomalies de logiciels, mais d'anomalie sur les fibres optiques elles-même, et de géolocaliser ces anomalies, pour y envoyer des agents pour réparer la fibre. Aussi pour des câbles réseau cuivre.

La solution était composée d'une partie logicielle, et d'une partie hardware: OS de type Linux, java, et langage C.

Duties and achievements:

- J'ai travaillé sur la partie logicielle, le "cerveau" de la solution: un serveur tournant dans JBoss EAP, Java JEE et Composants JCA (Java Connector Architecture).
- J'ai introduit le bare metal provisioning: avec du `PXE-` booting, pour déployer le stack complet, ce afin de pouvoir automatiser et rendre reproductible les tests.
- J'ai introduit et convaincu de la nécessité des "Acceptance Tests".
- J'ai développé un module d'authentification avec des Jee Listeners , et Spring pour la modularisation.
- J'ai introduit Docker pour déployer la solution, l'usage de Gitlab Pipelines avec un Gitlab que j'ai provisionné en privé dans nos serveurs.
- Pour la géolocalisation, j'ai introduit leaflet côté Web UI
- Cette expérience a été un moment capital de ma carrière: j'y ai travaillé avec un "grand maître Linux", **Eric Lévenez**, avec qui j'ai réellement appris ce qu'est un système d'exploitation.

IS Architect, IT department founder

06/2013 – 07/2015

Project Context: Département IT d'une PME de l'industrie du bâtiment, logiciel de gestion des tâches

Tech stack:

- *Operational:* Gantt charts, Specific Terms and Conditions documents in building industry (CCTP in French) / Response to tenders (« Réponses à appel d'offres » in French) / supply chain management on a construction site.
- *Devops:* Chef.io, Virtual Box SOAP API, KVM Virtual Manager, GNU/Linux (CentOS/Ubuntu server)
- *CI stack:* Artifactory, MAVEN3 (M2E), Git, Eclipse (eclipse bpmn modeler), Jenkins, Checkstyle, JMeter, JUnit, Mockito, DbUnit.
- *Dev stack:* Spring, JPA, Activiti BPMN engine, petals ESB / JBI java standard, tomcat 7, JAX-WS/ SOAP/ WSDL, JAX-RS, Spring WS, JSF2, Prime Faces, jQuery v2.x.x, JavaMail, Apache James server, Mailets.
- *Misc.:* **FM 200** Fire protection systems...?

SPVC était une PME du bâtiment, spécialisée dans les fluides.

J'ai conçu le premier véritable système d'information de l'entreprise, et ai participé au développement de composants logiciels nécessaires à ce système d'information

J'ai structuré cette petite DSI naissante, afin qu'elle soit en mesure de gérer un environnement de production. JE qualifierai cette expérience comme la plus folle de ma carrière: je n'avais pas conscience de ce dans quoi je m'engageais, en me lançant dans la construction d'une (même) petite DSI, en partant de zéro.

C'est par exemple dans ce contexte, que j'ai pour la première fois "travaillé avec de la vraie infra".

Duties and achievements:

- J'ai commencé par concevoir et implémenter un logiciel au départ assez basique, de gestion de tâches sur un chantier. Le but étant de normaliser des business process, en particulier pour simplifier la vie de la partie commerciale, et le suivi de projet.
- Je l'ai ensuite rendu plus mature pour la production, en y ajoutant un premier DRP. Les fonctionnalités étaient en relation avec le project management sur un chantier du bâtiment: Gantt Charts, Tasks Management, reporting.
- Une deuxième génération a donné un Système d'information de type B.A.M. (Business Activity Monitoring) utilisant :
 - Le moteur BPMN Activiti
 - Un ESB `JBI` couplé au moteur BPMN, une architecture SOA (l'exécution de `BPMN` était invoquée à travers l'ESB, via JAX-WS / JAX-RS).
 - 15 utilisateurs en Janvier 2014, et jusqu'à 30 fin 2014, puis 40 fin 2015.
- J'ai lentement construit une petite équipe, faite de 2 développeurs, et de 2 / 3 ops engineers (system engineer, security engineer, network engineer).

Java Dev Engineer

02/2013 – 05/2013

Project Context: Atermès, une société française affiliée à Thalès, industrie militaire.

Ingénieur de développement Java, J'ai travaillé dans le département R&D, où j'ai travaillé avec Luciad Lightspeed framework (un framework SIG, un pionnier de la 3D dans le domaine des SIG).

Duties and achievements:

- Conçu et implémenté un ensemble de composants Java JSE6 Swing, en utilisant le framework Luciad Lightspeed, et le concept d'analyse hypsométrique avec les "IndexColorModels".
- J'ai amené l'utilisation de tests Junit, l'équipe travaillait surtout avec le framework UI Qt C++.
- J'ai beaucoup appris à propos des SIG:
 - C'est après cette expérience, que je me suis intéressé à OpenStreetMap, Leaflet, etc.
 - Concepts fondamentaux: géoïde, ellipsoïde de référence, système de référence, projections, point de référence, datum, serveur de tuiles, "line-of-sights", etc....

Tech stack:

- *OS:* Windows, on Dell Alienware.
- *Devops:* Chef.io, Virtual Box SOAP API, KVM Virtual Manager, GNU/Linux (CentOS/Ubuntu server)
- *CI stack:* Maven + Junit.
- *Dev stack:* Swing JSE6, JAXB, Luciad Lightspeed, MVC2 design pattern.
- *Misc.:* a bit of differential geometry of finite dimensional manifolds.

Dev Engineer

01/2012 – 12/2012

Project Context: DSI Michael Page.

Tech stack:

- *CI stack:* Maven, Tortoise SVN, JIRA, Jenkins, Nexus Sonatype.
- *Dev stack:* Tomcat 6, WebLogic 9, struts 1, Taglibs, Spring, hibernate, JUnit, JMX, JAXP, Log4J, Fractal Julia, Javamail.

Au cœur de la DSI Monde Michael Page, zone EMEA, j'ai pris part à la refonte complète du Système d'information avec 10 à 15 autres ingénieurs de développement java, 2/3 Architectes IT, un QA et 10 Business Analysts.

J'étais dédié à la partie « backend » des composants relatifs à la gestion de l'évaluation des candidats (ceux qui viennent faire un entretien de recrutement chez Michael Page. Ce fut ma première expérience dans une DSI de grande taille.

Duties and achievements:

- En collaboration avec les Architectes, et Business Analysts, j'ai conçu et implémenté plusieurs composants impliqués dans le "candidate assessment". Je suis devenu SME pour le fonctionnel "Candidate Assessment".
- Dev Java classique, avec CI, Maven, JUnit, Nexus Sonatype.
- J'ai eu l'occasion de prendre part à une opération de sécurité suite à une attaque subie par le groupe un souvenir mémorable de week-end pizza passé au bureau
- J'ai commencé à être impliqué dans des tâches de CI/CD avec l'infrastructure qui va avec (un serveur Jenkins).

{ 2011: I spent one year in a French IT school named « AFCEPF » to get a Software Architect degree }

> *Sanofi Aventis*

Java Engineer

02/2010 – 12/2010

Project Context: At Sanofi Aventis, R&D, Genetics Data Analysis

Tech stack:

- *CI stack:* Subversion, Tortoise SVN.
- *Dev stack:* JavaEE Servlet Containers, Struts 2, Taglibs, Tomcat, Spring, Hibernate, Junit.

Freelance au sein d'un département de R&D chez Sanofi Aventis, j'ai été engagé pour concevoir et implémenter un logiciel capable de collecter, et analyser pour produire des dashboards de résultats, dans le domaine de la génétique: Les chercheurs avaient pour but d'expérimenter l'utilisation de techniques de data-mining.

The main requirements were:

- The software modularization: It should be easy to add more analysis modules, such as within a plugin model: The researchers wanted the ability to add custom analysis modules on demand.
- Moreover, the plugins must not require any change for deployment, except editing a configuration file, and a restart.

Duties and achievements:

- Java, analyse UML,
- Architecture MVC2, struts 2, Spring, Hibernate
- Configuration et Setup des environnements de développement: Eclipse , Maven, Nexus, SVN Subversion et ses plugins eclipse/tortoise.
- J'ai conduit des workshops pour former 2 chercheurs au développement en java, avec Struts2, Spring et Hibernate: ils souhaitaient être en capacité de développer eux-même leur propre modèles d'analyse, avec l'aide de consultant lorsque nécessaire, mais en gardant un tronc de connaissance en interne.
- J'ai été initié à quelques concepts d'épi-génétique

Java Engineer SME

03/2009 – 11/2009

Project Context: Logiciel de Gestion des tâches en Java

Tech stack:

- *CI stack:* Tortoise SVN, Junit, Ant.
- *Dev stack:* Java Swing, Spring JUnit, Tom, Axis 2, Tomcat, Spring, Hibernate, JUnit, Netbeans, Ant

J'ai été engagé en tant que freelance, avec le rôle de SME Java, dans une équipe qui développait un logiciel de gestion de tâches.

Le logiciel était fait avec Java Swing, avec une communication côté serveur à base de web services soap (tomcat axis2), et une BDD Microsoft SQL Server.

All classical public administration needs, nothing fancy.

Duties and achievements:

- Training the Devs to Java Swing, Spring, Hibernate Frameworks
- UML / Use Cases based analysis
- Java Dev (I coded, I always get bored if i do not code), And was required to give my point of view for toics where the team members needed Java Tech expertise.
- I introduced Junit Testing, and the concept of "acceptance tests" mirrored on Use Cases.

> Mnémosyne

.NET Developer

10/2008 – 12/2008

MNEMOSYNE était un éditeur de logiciel pour les notaires, le logiciel devait permettre de faire de la Gestion électronique de documents (GED), les documents devant pouvoir être authentifié par l'état.

I there developed a C# ASP.NET search field auto-completion module.

Tech stack:

- *Dev stack:* Visual Studio, C# .NET, ASP .NET Ajax, Telerik ASP.NET framework.

Duties and achievements:

- J'ai développé un composant C# ASP.NET d'auto-complétion, qui permettait la recherche de clients basé sur des critères tels que le nom, ou l'adresse.
- J'ai appris à utiliser le framework Telerik Ajax pour ASP.NET.

> Ascott

Java & .Net Developer

06/ 2008 – 07/ 2008

Ascott était une société de consultant IT de taille moyenne, mon premier employeur.

J'ai travaillé pour leur client principal: la société française SCOR, acteur historique de la ré-assurance.

J'ai réalisé quelques tâches de développement ASP.NET, et ai passé la plupart de mon temps à implémenter des "Crystal Report", dans un projet de Business Intelligence.

Tech stack:

- *Dev stack:* Crystal Reports XI, ASP .NET ADO.NET, et IngresDB en .NET.

Duties and achievements:

- Design / Configuration de « Crystal Report » (requêtes SQL + mise en forme), intégration au Système d'Information. Les Reports étaient intégrés avec le logiciel de Business Intelligence « SIMCORP Dimension »
- J'ai développé quelques pages ASP.NET avec ADO.NET et un driver ODBC spécifique pour aller requêter des données dans un très vieux SGDBD appelé « Ingres »,
- J'ai été initié aux standard IFRS , à la Business Intelligence, et à ce qu'est le métier de la ré-assurance.

2011	Titre Architecte Logiciel	AFCEPF, http://www.afcepf.fr/
2007	Formation développeur, équivalent BAC+2	AFPA Paris, https://www.afpa.fr/
2001	DEUG MIAS (Mathématiques)	Paris 6 University, http://www.upmc.fr/en/
1999	DEUG SCVT, (Biochimie)	Paris 6 University, http://www.upmc.fr/en/

Langues

Anglais courant, Allemand scolaire, quelques mots de Bulgare, quelques mots de tagalog.

Et...

Jogging, randonnée, escrime (épée et sabre), pêche à la truite (j'adore la pêche à la truite :)).