



# Payroll as Simple as Pushing a Button



## COMPANY BACKGROUND

Salarium is a SaaS end-to-end payroll automation company that provides enterprise payroll system software for small, medium and large businesses. It is headquartered in Hong Kong and has an office in Manila, Philippines.

## Project Overview

Exist helped Salarium automate its software development lifecycle from development to testing to production through setting-up container orchestration and CI/ CD tools. To achieve faster time to market

## Challenges

- Salarium launched a product in the market and instabilities - issues, defects, and problems transpired post-launching. They wanted to create a new version in an attempt to address the instabilities.
- With a fully distributed team working across different time zones and platforms, Salarium was looking to solve its current architecture -- using microservices but wasn't running in containers yet. Resulting in difficulty in maintaining the architecture, longer release cycles, and lack of scalability/agility.
- They wanted to containerize their applications and deploy orchestration and automation tools so they can improve their delivery pipeline and manage their app more reliably across different environments. However, the company lacks the proper resources to effectively implement the cloud-native approach.



## Solution

- Salarium entrusted its technological dilemma to Exist believing the latter's capability to build the appropriate solution.
- Exist implemented the container orchestration and CI/CD tool to automate the deployment and release cycle. The team also provided consulting services to Salarium to secure that Salarium is able to offer a steady solution to its clients.
- Exist created DC/OS cluster with three (3) Kubernetes environments under it - development, staging, and production.

## DevOps Implementation with Container Orchestration

Exist provided cluster management for Kubernetes This enables DevOps opportunities for our client. In fact, they have accomplished CI/CD practices incorporating automation pipelines using GitLab, Jenkins and Kubernetes Ingress, built from the Rancher UI. Developers may now test their container implementation in a Kubernetes testing namespace with a single Git push, while only operators will be able to deploy containers in the production namespace.

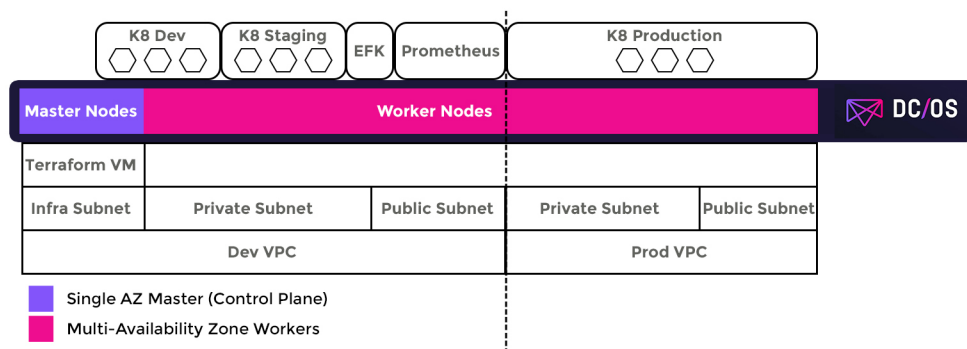


Figure 1: Architectural Design of the Project

- 3 developers worked for the project in 2 months
- Exist tailor-fitted the solution having the necessary features for the operational needs of the client. It used several technologies such as:
  - **Docker** - to containerized the applications
  - **DC/OS Mesosphere** - platform
  - **Kubernetes** - container orchestration system
  - **GitLab CI/CD** - configured the enterprise version and integrated it to Kubernetes
  - **Elastic logs, Elastic Logstash Kibana (ELK) stack and Grafana** - monitoring tools
  - **Terraform** - to create the managed Kubernetes cluster

## Results

### Business Benefits

- The process became faster. The company was able to adapt as the cycles are working well just as how they envisioned it.

### Technical Benefits

- There has been an assurance that the release had been deployed to different environments and tested.
- It can use different versions and deploy different applications
- Their CI/CD process improved. It also improved the way they set up applications, especially microservices architecture and utilized the features more efficiently as well as monitoring tools.
- It solved issues in management, auto-spawn
- Improved development; each team can now develop on their own
- There has been harmonious integration despite varying programming language
- Improvements in the delivery
- The client plans to engage with Exist again for other services such as QA automation, optimization of architectural design and components upgrade.