

Caso de estudio Nro 2- Cablemedia

La Empresa se llama Cablemedia es una compañía regional, de gran tamaño, que opera en Argentina, Uruguay y Paraguay. Está en el rubro de televisión por cable, telefonía fija digital e Internet para minoristas y clientes corporativos.

Su misión es “proveer soluciones de calidad y de vanguardia, a las necesidades de comunicación digital y de entretenimiento”. Brinda soluciones dirigidas a clientes minoristas y corporativos, con adaptaciones, personalizadas a cada cliente. Tiene alrededor de 1000 empleados.

Cada división tiene los siguientes departamentos: Recursos humanos: 20 empleados, Coordinación con mercadotecnia: 30 empleados, Operaciones: 160 empleados, Suministros: 20 empleados, Contabilidad y control: 20 empleados

En la casa matriz, están los siguientes departamentos: Administración: 30, Mercadotecnia: 50, Finanzas: 70, Producción y control: 100 empleados.

La empresa Cablemedia tiene la necesidad de incorporar un motor de reglas a su aplicación web de gestión interna de catálogos de ofertas, promociones, productos y precios comerciales. Con el fin de aumentar las ventas, y por lo tanto también aumentar el beneficio.

Para ello decidió segregar su actual sistema productivo monolítico en microservicios. Los clientes se comunicarán con la aplicación principal donde estarán expuestos los servicios con api restful de consultas/altas/bajas y modificaciones, con su base de datos relacional (Mysql) y esta a su vez se comunicará mediante otra api que expone la aplicación donde se encuentran almacenadas las reglas en una base no relacional (Mongodb) y las retornará a la primera aplicación, para que ella toma la decisión de qué productos ofrecerle al cliente de la aplicación web en función de lo que resuelva el motor de reglas. Se decidió este enfoque ya que la aplicación que expone, guarda, modifica y da de alta reglas es consultado por un conjunto de aplicaciones adicionales, además de la de gestión de catálogos. Se decidió que el área de desarrollo e implementación de la compañía clusterice las aplicaciones y sus correspondientes bases de datos operativas, en un hosting en la nube basado en Kubernetes y Docker de IBM con el debido SLA, ya que no se justifica hacer una inversión para montarlas en “on-premises”. Sin embargo, los datos más sensibles de los clientes se almacenarán en un sistema clusterizado “on-premises” aparte, en la casa matriz, que servirá de consulta y resguardo ante necesidades administrativas y legales. En particular, también contendrá el catálogo de ofertas, promociones, productos y precios comerciales por cada división. También estará basado en Kubernetes y Docker, usando arquitectura multiservicio con APIs Restful, y una base de datos relacional.

De ésta manera, el riesgo de acceso no autorizado a la información personal identificable se reduce y se mitiga, ya que la información PII más sensible no queda accesible a la compañía que provee el servicio de Cloud Hosting, y a sus posibles intrusos ante un compromiso eventual. En caso de que quede el hosting comprometido, sólo queda divulgada información PII estrictamente necesaria para la operación del primer sistema mencionado.

Por la magnitud de la empresa que provee el hosting, se asume que el riesgo de no disponibilidad está suficientemente tratado. En caso de necesidad, la migración a otro hosting es altamente viable, gracias a Kubernetes y al sistema “on-premises” propio de la casa matriz.