

Portafolio del Mes - Octubre

Autor: Vicente Fuenzalida Marín

Fecha de Entrega: 13 de octubre de 2017

Tema: Cloud Computing

Ventajas de Cloud Computing

Artefacto

El artefacto que escogí esta semana es una [página informativa](#) del sitio LevelCloud donde se enumeran las ventajas y desventajas de este paradigma frente a la forma anterior de utilizar servicios (localmente).

Las ventajas que se mencionan son las siguientes:

1. **Ahorro en Costos:** Se paga solo por lo que se usa, lo que implica ahorro significativo ya que no se deben hacer grandes inversiones para poder escalar, y cuando haya menor carga de trabajo simplemente se puede disminuir la cantidad de recursos utilizados (si fuesen propios, la tasa de ocupación de recursos sería deficiente en muchos casos). Además, se ahorra en costos operacionales y de mantención (infraestructura, administración, energía, etc).
2. **Confiabilidad:** Los proveedores ofrecen SLA (acuerdos de nivel de servicio) que garantizan disponibilidad del 99,99% todos los días del año. Además manejan recursos redundantes y mecanismos en caso de fallas (pueden mover sus servicios fácilmente de un servidor a otro).
3. **Manejabilidad:** Se simplifica la administración TI y gestión de los recursos ya que los proveedores se encargan de la mantención de sus recursos.
4. **Ventaja Competitiva:** Provee ventaja sobre los potenciales competidores (del cliente), ya que la obtención de nuevos recursos es prácticamente inmediata (escalabilidad). Permite abstraerse y 'olvidarse' de toda la gestión tecnológica para enfocarse en las actividades y objetivos comerciales de la empresa/cliente. Reduce además el tiempo necesario para llevar un producto (aplicación o servicio) al mercado.

Las desventajas son las siguientes:

1. **Downtime:** Debido a que los proveedores de servicios Cloud deben manejar a múltiples (y muchísimos) clientes continuamente, pueden eventualmente saturarse y provocar la caída de una aplicación/servicio

temporalmente. Además, siempre será necesario tener conexión a internet para acceder a las aplicaciones de las que se es dueño para poder administrarlas.

2. Seguridad: Guardar datos y archivos utilizando proveedores externos siempre representará riesgos, ya que se le está otorgando acceso al proveedor tanto a la lógica de negocios (aplicación misma) como a datos delicados. Al ser un servicio "de terceros", también se abre la posibilidad de que agentes maliciosos encuentren fallas y aperturas para acceder a los servicios Cloud provistos.
3. Dependencia del Proveedor: Puede resultar muy difícil migrar de un proveedor a otro debido a problemas de interoperabilidad y soporte.
4. Control Limitado: El control que tienen los clientes de los servicios Cloud es mínimo ya que todo es administrado por los proveedores. Por esto, los clientes no pueden manejar trabajar sobre la infraestructura libremente, actualizar firmware, ni acceder a las terminales de los servidores, labores muy comunes en la tenencia y mantención de una aplicación/servicio web.

Reflexión

A partir de la evidencia analizada, pude notar que "no todo es tan bueno" con respecto al uso de servicios Cloud, y que cada vez que se comience el desarrollo de un proyecto se debe tener en consideración muchos otros aspectos que dependerán únicamente de las necesidades y requisitos levantados.

Por ejemplo, para algún proyecto de desarrollo de software podría ser innecesario mantenerlo online si su uso no excede lo *local* (aplicaciones de uso en una red cerrada por ejemplo). También, si los servicios no son directamente compatibles con las necesidades y el problema a resolver, es más complejo adaptar la solución a los servicios existentes (forzar la compatibilidad) que manejarlo de manera interna.

Como se lista en la evidencia escogida, el hecho de *ceder* un poco el control de un componente/funcionalidad de la aplicación a un agente externo puede ser complejo debido a la pérdida de control tanto en términos de configuración como de seguridad, lo que podría dejar intranquilos a los desarrolladores si los datos o funcionamiento deben ser privados. Sin embargo, dentro de los principios y consejos que se señalan en todas partes siempre está el de **no reinventar la rueda**, y de **confiar en artefactos testeados y validados por la comunidad**.

A mi al menos me queda la sensación de que **no hay que temerle** a estos servicios, ya que están respaldados por equipos de desarrolladores y grandes empresas, y además su uso es extenso dentro del ecosistema. Por otro lado, favorecen la abstracción y simplificación de muchas tareas, lo que ahorra tiempo al momento de desarrollar, y facilidad a la hora de monitorear y mantener las aplicaciones resultantes.

He utilizado algunos de estos servicios personalmente (AWS y Heroku), y mi experiencia ha sido bastante satisfactoria, ya que realmente **simplifican mucho las tareas** de deployment y otras (escalamiento

automático, integración continua, integración con otros servicios, testing, etc). Lo único que se debe tener en mente siempre es que no existe una solución que siempre sea *buena* o *mala* para todos los proyectos, sino que las tecnologías a escoger deben estar siempre fundamentadas según los requisitos y restricciones del problema a abordar.