Computación en la Nube: Análisis de Proveedores y Servicios para una Aplicación de Ciudades Inteligentes

<<Nombre y apellidos – Autor1>><<Nombre y apellidos – Autor2>>

<<Datos de filiación – Autor 1>><<Datos de filiación – Autor 2>>

<<Email Autor1>><<Email Autor2>>

Resumen

El presente proyecto detalla el estudio realizado de proveedores de computación en la nube (cloud computing) y los servicios ofrecidos por ellos. Este estudio se fundamente en el desarrollo de una aplicación de Internet de las cosas en el entorno de una ciudad inteligente. Luego de realizada una introducción a ciudades inteligente y la importancia de estas, se detallarán los tres tipos de servicios básicos que los proveedores pueden ofrecer, se planteara una aplicación a implementar, y sobre la base de esta se analizaran los servicios de los principales proveedores de computación en la nube.

Introducción

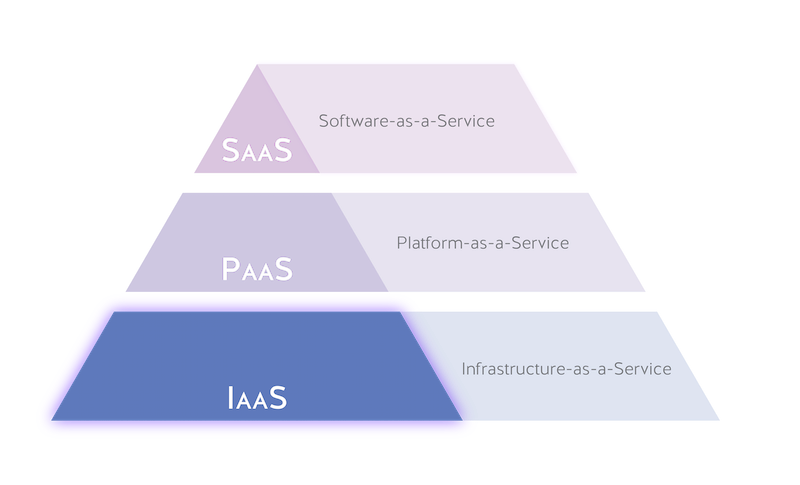
Las ciudades modernas son las responsables del 80% de la producción económica mundial y del 70% del consumo de energía mundial y las emisiones de gases de efecto invernadero, y, para el año 2050 serán el hogar del 66% de los 9.000 millones de habitantes proyectados. Por lo tanto, es necesario pensar en políticas que permitan la sustentabilidad de las ciudades [5]. Las ciudades sustentables deben mejorar la calidad de vida de los habitantes, la eficiencia de las operaciones y los servicios urbanos, y su competitividad, mientras se asegura que esta se encuentre de acuerdo a las necesidades de las generaciones presentes y futuras en lo que respecta a aspectos económicos, sociales, medioambientales, así como también culturales. Las tecnologías de la información y las comunicaciones, y en especial Internet de las cosas, tienen un papel muy importante, estas actúan como una plataforma para obtener datos e información que ayudará a mejorar en entendimiento sobre como la cuidad está funcionando en termino de consumo de recursos, servicios y calidad de vida. Como ejemplo de los problemas cotidianos que se intentan solucionar con estos dispositivos podemos citar la potencial aparición mosquitos a raíz de un aumento en la temperatura y humedad en un parque, la cual puede ser detectada por sensores y generar un proceso de fumigación [6]. Otro ejemplo es la red de monitoreo de calidad de aire y ruido de la ciudad de Buenos Aires, la cual permite controlar la calidad ambiental del entorno urbano a través de un seguimiento continuo de los niveles de contaminación y así generar información confiable, comparable y representativa para su aplicación en la estrategia local de protección de la salud y el ambiente [7].

Tipos de servicios de computación en la nube

Las aplicaciones de Internet de la Cosas, y dentro de ellas las ciudades inteligentes, están formadas por varios elementos: los dispositivos finales, lo que vulgarmente llamamos cosas, que son los que generan los datos; los Gateway o Routers de datos quienes envían los mismos hacia la nube; la infraestructura, y finalmente los servicios y aplicaciones.

La computación en la nube viene a cubrir las necesidades planteadas en los últimos dos elementos. La misma, es un paradigma que permite ofrecer servicios de computación a través de una red, que usualmente es Internet. La computación en la nube son servidores en Internet encargados de atender las peticiones en cualquier momento. Se puede tener acceso a su información o servicio, mediante una conexión a Internet desde cualquier dispositivo móvil o fijo ubicado en cualquier lugar. Esta medida reduce los costos, garantiza un mejor tiempo de actividad y la invulnerabilidad de los sitios web a los [delincuentes informáticos](https://es.wikipedia.org/wiki/Delito_inform%C3%A1tico), a los gobiernos locales y a sus redadas policiales pertenecientes.

El concepto de “nube informática” es muy amplio, y abarca casi todos los posibles tipo de servicio en línea, pero cuando las empresas ofrecen un servicio alojado en la nube, por lo general se refieren a alguna de estas tres modalidades: el [Software como Servicio](https://es.wikipedia.org/wiki/Software_como_servicio) (por sus siglas en inglés SaaS -Software as a Service), Plataforma como Servicio (PaaS) e Infraestructura como Servicio (IaaS), la figura N°1 presenta la relación entre ellas. El software como servicio (SaaS) es un modelo de distribución de software en el que las aplicaciones están alojadas por una compañía o proveedor de servicio y puestas a disposición de los usuarios a través de una red, generalmente la Internet. Plataforma como servicio (PaaS) es un conjunto de utilitarios para abastecer al usuario de sistemas operativos y servicios asociados a través de Internet sin necesidad de descargas o instalación alguna. Infraestructura como Servicio (IaaS) se refiere a la tercerización de los equipos utilizados para apoyar las operaciones, incluido el almacenamiento, hardware, servidores y componentes de red.

 Figura 1. Estructura de tipos de servicios en la nube.

Estacionamiento inteligente

Implementaciones a nivel mundial

Debido al crecimiento de la población urbana en las grandes ciudades, el parque automotor ha aumentado considerablemente, causando una serie de inconvenientes en el desplazamiento vial. Si bien la posesión de un vehículo brinda a los ciudadanos la facilidad de viajar, muy importante en las grandes ciudades donde los destinos son distantes, el aumento de la cantidad de autos en circulación tiene un impacto negativo en la calidad de vida de los ciudadanos por varios motivos, entre los cuales se encuentra la pérdida de dinero por extensas búsquedas de estacionamiento y el consumo de combustible, un empeoramiento en la calidad del aire y contaminación ambiental debido al tiempo extra de circulación de los vehículos, pérdida de tiempo y disminución de productividad por la cantidad de horas desaprovechadas por los automovilistas [16].

Muchas ciudades han mejorado estos problemas con el uso de parquímetros que permiten realizar un control de los espacios de estacionamiento y mejorar la movilidad en la ocupación de esos espacios, logrando una rotación de los lugares ocupados, por ejemplo: en la ciudad de San Francisco, en Estados Unidos, con la implementación de parquímetros inteligentes se ha logrado reducir en un 50% los tiempos búsqueda de lugar de estacionamiento mejorando tanto la calidad de vida del usuario como la contaminación ambiental[17]; en la ciudad San Luis Potosí, en México, la implementación de un sistema ha logrado mejorar la disponibilidad de lugares de estacionamiento y reducir el número de coches que se estacionan ilegalmente, en un 70% y mejoro el medio ambiente, logrando reducir las emisiones contaminantes de los autos al lograr un estacionamiento en tiempo reducido [18]; en la ciudad de Moscú, Rusia, el sistema desplegado por la empresa Worldsensing [22] logró reducir en gran medida la congestión de tráfico en zonas críticas y con ello los niveles de contaminación ambiental y estrés de los conductores; y en el ámbito local en la ciudad de Bahía Blanca, Prov. de Bs.As., se instaló una nueva plataforma de estacionamiento inteligente y un conjunto de servicios tecnológicos que, combinados entre sí, mejoran notablemente los tiempos de interacción entre el usuario y el uso del sistema de estacionamiento medido[23].

Definición de nuestra aplicación

A continuación, se hace un análisis de los distintos escenarios para implementar un estacionamiento inteligente en el ámbito de la Universidad y se plantea los posibles servicios de computación en la nube que pueden ser necesarios.

**Ingreso**

Los puntos relevantes al ingreso son:

1. Cada vehículo/usuario cuenta con una identificación, la misma puede ser realizada a través de un tag de RFID/NFC o a través de una aplicación celular.
2. Un dispositivo lector tomara la identificación del vehículo/usuario.
3. Dicha información se compara con los distintos ID de los vehículos/usuarios almacenados en una base de datos, la cual puede estar en la nube.
4. Si el ID es encontrado la puerta se abre.

En el punto 3 la identificación obtenida debe ser manipulada por una base de datos, preferentemente en la nube para poder acceder a ella desde cualquier ubicación. Se deberá poder acceder a la información cuando sea necesario para poder hacer estadísticas gráficos y consultas de los ingresos al establecimiento, horarios más concurridos entre otros. Para el cual, se necesita una infraestructura en la nube capaz de almacenar los datos históricos, y otro capaz de poder consultarlos cuando sea necesario. Vale aclarar que la base de datos deberá tener la seguridad correspondiente para que los datos no sean violados ni alterados.

**Disponibilidad**

Los puntos más importantes asociados a la disponibilidad de las plazas de estacionamiento son:

1. Una vez que el vehículo/usuario ingrese al estacionamiento, el conductor desde el dispositivo móvil visualiza los lugares de estacionamiento disponibles.
2. Éste selecciona un lugar para estacionar su vehículo, pudiendo indicar el tiempo que estará allí, y estacionará.
3. Cuando el vehículo deje de utilizar la plaza. Pasará su estado de “Ocupado” a “Disponible” en la aplicación para que otro usuario pueda usarlo.

Para este segundo escenario el sistema deberá operar con bases de datos en tiempo real (donde los datos serán consultados con mayor frecuencia, para ver el estado de la plaza). Esto implica que el sistema trabaje con servicios online para poder establecer los tiempos de cada plaza y su estado (ocupado o disponible) en el instante. Ya que, dependiendo del tamaño del estacionamiento cambiará constantemente. Entonces, es necesario el uso de un servicio base de datos para poder tener un control preciso del estado de la plaza.

**Pago**

A continuación, se detallan los puntos relevantes asociados al pago del estacionamiento, el cual se realiza al momento de retirarse del estacionamiento:

1. El conductor podrá contener dentro de la aplicación una pestaña de pago “billetera digital” que podrá cargar tanto en kioscos como con tarjetas de débito y crédito. El sistema podrá contemplar otros métodos de pago.
2. El cobro dependerá del tiempo y el vehículo en cuestión.
3. Al ingresar al establecimiento conductor recibirá un comprobante, pudiendo ser el mismo virtual a través de la aplicación móvil. Al momento de retirase se chequera el tiempo que estuvo por medio del comprobante y el conductor podrá realizar el pago a través de la aplicación o a través de otro medio.

En este escenario se deberá trabajar con sistemas de tiempo real de cobro, que permita mantener informado al usuario de su estado de cuenta y al estacionamiento del monto que posee el mismo para efectuar el pago.

**Beneficios**

El conductor podrá recibir beneficios. Si es discapacitado podrá tener lugares privilegiados, o si es un cliente habitual podrá recibir promociones o descuento. Para la implementación de esto, tan solo debe tener en cuenta un campo extra en la base de datos.

**Administrador del estacionamiento**

El administrador del estacionamiento debe tener conocimiento de estado y control de cada plaza del estacionamiento, logrando tener estadísticas de días con mayor concurrencia, tiempo de estadía y horas por vehículo. Es de suma importancia, que el administrador pueda tener un correcto acceso a la base de datos de su estacionamiento y corroborar los ingresos que le genere. Por otro lado, podrá acceder a estadísticas semanales mensuales y anuales, para ello es necesario que el administrador tenga acceso a la interfaz de administración, donde el sistema guardará los reportes semanales, mensuales y anuales, y le permita cambiar las configuraciones del sistema, con gráficos y estadísticas para que pueda visualizar la productividad del estacionamiento.

**Conclusión**

Como resumen de los escenarios vistos, se llega a la conclusión que el sistema, independiente al hardware asociado, deberá poseer un sistema de base de datos con suficiente capacidad para el almacenamiento del historial de datos del estacionamiento, y así poder realizar la administración del ingreso, la disponibilidad de plazas, el cobro, la generación de beneficios y la realización de estadísticas que permitan un uso más eficiente del estacionamiento. Si la aplicación de gestión se pretende que este disponible en la nube, se deben considerar la disponibilidad de las herramientas para el desarrollo de esta.

Por lo tanto, se debe pensar en contratar servicios de IaaS, los cuales están asociados a los servidores, el almacenamiento y a las bases de datos que se emplearan. Así como también servicios de PaaS para el desarrollo de la aplicación de gestión en la nube.

Prestadoras de servicios

Son muchas empresas hoy en día que se dedican a prestar servicios de computación en la nube tales como Microsoft, Amazon, IBM, Google, Alibaba, entre otras. A continuación, se analizar las soluciones de tres empresas: Microsoft Azure, Amazon Web Services e IBM Cloud

Si bien en la sección “Tipos de servicios de computación en la nube” se detallaron los mismos, cuando vamos a la web de los proveedores seleccionados nos encontramos con escenario que no es el esperado. Encontramos una gran variedad de productos organizados de otra forma, las figuras N°2, 3 y 4 presentan una imagen de dicha páginas web [][][]. Dentro de estos productos encontramos:

1. Computación
2. Almacenamiento
3. Base de datos
4. Redes
5. Multimedia
6. Herramientas del desarrollador
7. Internet de las Cosas
8. Blockhain
9. Y muchos otros más.

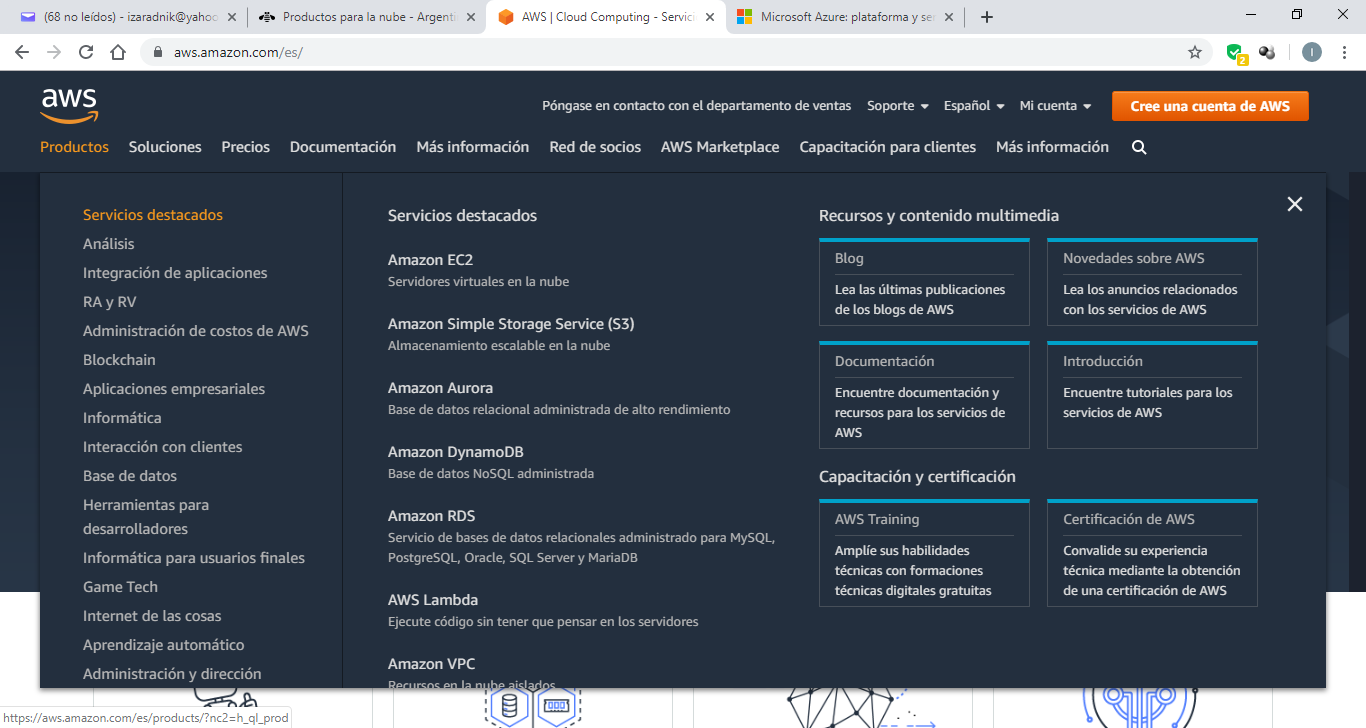


Figura 2. Página Web AWS - Productos.

Imagen que contiene captura de pantalla

Descripción generada automáticamente

Figura 3. Página Web Azure - Productos.

Imagen que contiene captura de pantalla, monitor, carretera, interior

Descripción generada automáticamente

Figura 4. Página Web IBM - Productos.

Si bien, tal como se detalló, todos los proveedores poseen productos asociados a Internet de las Cosas, los cuales integran a su vez productos de almacenamiento y bases de datos, se comenzó haciendo un análisis de los productos base para lograr un mejor entendimiento y una selección más criteriosa. A continuación, se presenta el análisis de los productos de servidores, bases de datos, almacenamiento y las herramientas para el desarrollador.

**Servidores**

Si bien estamos hablando de un servicio en la nube, este debe estar soportado por algún tipo de hardware. Por lo tanto, se debe considerar el tipo de hardware que poseen las instalaciones de las prestadoras de servicios como la de base datos, a fin de poder determinar el óptimo para nuestra aplicación.

Para poder comparar tomamos a modo de ejemplo 3 prestadoras de servicios. Como mencionamos anteriormente estaremos analizando Microsoft Azure, Amazon Web Services e IBM Cloud.

**Tabla 1. Servidores**

|  |  |  |
| --- | --- | --- |
| Microsoft Azure | Amazon Web Services | IBM Cloud |
| Gen 4 | Económico | Estándar Pequeño |
| Gen 5 | Premium | Estándar Mediano |
|  |  | Estándar Grande |

En el cuadro anterior se puede apreciar los diferentes tipos de paquetes que ofrecen cada prestador de servicios, donde los servicios están ordenados de menor a mayores especificaciones.

**Microsoft Azure**

El primero Microsoft Azure que ofrece dos alternativas la más económica “Gen 4” con procesadores Intel de 2x 2,4 GHz variando en sus Capacidades pudiendo elegir entre 56GB, 112GB y 157 GB de memoria, Mientras que la versión “Gen 5” ofrece procesadores Intel de 4x 2,3 GHz y sus capacidades de almacenamiento varían en 40.8GB, 81.6GB, 122.4GB, 163.2GB, 204GB, 326.4GB, 396GB de memoria [1]

**Amazon Web Services**

Por otra parte, Amazon Web Services también se dividen en dos versiones, la versión “económica” que posee un núcleo de 1 ACU con 2 x 2.0 GHz, 8 Gb de RAM con una capacidad de 150GB de almacenamiento, y una versión “Premium” que posee un núcleo de 4 ACU con 8 x 2.0 GHz, 32 Gb de RAM con 500GB de almacenamiento [2]

**IBM**

Y por último IBM tiene tres categorías, que están divididos en “Estándar pequeño” núcleos de 2 x 2.0 GHz, 8GB de RAM, 1x100GB, con 1x500GB de almacenamiento, “Estándar Mediano” con núcleos de 4 x 2.0 GHz,16GB de RAM,1x1TB de almacenamiento y por ultimo “Estándar Grande” con núcleos privados de 8 x 2.0 GHz, 32GB de RAM, 1x100GB (SAN), 1x2TB de almacenamiento. [3]

**Conclusión**

Se puede observar que los tres ofrecen alternativas en sus productos, una versión Premium y otra más reducida, pero en el caso de IBM se tiene además una versión media, entre la Premium y la reducida, que nos da más opcionalidad a la hora de elegir el proveedor y el tipo de producto que terminaremos eligiendo que se adapte lo máximo posible a nuestras necesidades.

**Base de Datos**

Independientemente del software a desarrollar tendremos una interacción con datos, por lo tanto, haremos uso de una base de datos para poder ser manipulados a continuación analizaremos los factores principales a tener en cuenta, que son seguridad y eficiencia

**Microsoft Azure**

//

**Amazon Web Services**

**//**

**IBM**

**//**

**Almacenamiento**

Si queremos que el sistema de software que estamos implementando guarde una gran cantidad de datos o necesitamos almacenar historiales de datos, debemos considerar implementar un servicio de almacenamiento de datos, para seleccionar el producto indicado hay factores a tomar en cuenta, como es el tamaño máximo de archivos que puedo almacenar, y la rapidez con las que puedo acceder a ellos que se mide en IOPS[[1]](#footnote-1)

**Microsoft Azure**

**//**

**Amazon Web Services**

**//**

**IBM**

//

**Herramientas de desarrollo**

Si el software a desarrollar lo necesita, se pueden incorporar herramientas de desarrollo,

**Microsoft Azure**

**//**

**Amazon Web Services**

**//**

**IBM**

//

Referencias

[1]. Azure,” Sql data base specification” https://azure.microsoft.com/es-es/pricing/details/sql-database/managed/ page official, 2018

[2]. Amazon Web Services. “MySQL data base specification” https://aws.amazon.com/es/rds/mysql/pricing/,page official,2019

[3]. IBM, “IBM database” https://www.ibm.com/analytics/db2. page official,2019

[3] Marcus, A., Semantic Driven Program Analysis, Kent State University, Kent, OH, USA, Doctoral Thesis, 2003.

[4] Marcus, A. and Maletic, J. I., "Recovering Documentation-to-Source-Code Traceability Links using Latent Semantic Indexing", in Proceedings 25th IEEE/ACM International Conference on Software Engineering (ICSE'03), Portland, OR, May 3-10 2003, pp. 125-137.

1. ***IOPS (Input/output Operations Per Second)***

   [↑](#footnote-ref-1)