



# COMPONENTES Y MODELOS DE SERVICIOS

DAW 2 – COMPUTACIÓN EN LA NUBE

PEDRO ARZA, JUAN DAVID SERVELLÓN Y  
ALEJANDRO SAINZ

## Indice de contenido

IaaS, PaaS, SaaS .....	2
IaaS : Infrastructures as a Service .....	2
PaaS : Platform as a Service .....	3
SaaS : Software as a Service.....	3
Tipos de Nubes .....	4
Nubes Públicas .....	4
Nubes Privadas .....	4
Nubes Híbridas.....	5
Ejemplo de Negocio .....	6
Propuesta de Negocio.....	6
Soluciones comerciales en la nube – niveles.....	6
Servicios que ofrecen más costes.....	6
Ventajas y desventajas de cada solución .....	7
AWS:Ventajas .....	7
AWS:Desventajas .....	8
Heroku : Ventajas.....	8
Justificación de elección.....	8
Diagrama de interacción entre los niveles de servicio .....	8

## IAAS, PAAS, SAAS

En este primer apartado comenzamos hablando sobre elementos como servicio, es decir, cualquier elemento de hardware o software que es ofrecido a clientes o particulares para que ellos puedan hacer uso de los mismos sin necesidad de preocuparse por la propiedad de los mismos. Basándonos en la propia jerarquía de este tipo de servicios vamos a enumerar 3, desde aquellos servicios que se enfocan en un todo hasta aquellas partes más livianas o que sólo engloban una aplicación.

### IaaS : Infrastructures as a Service

Los usuarios tienen un mayor control sobre la infraestructura, lo que les permite personalizarlas según sus necesidades, aunque requieren conocimientos técnicos especializados para su configuración y administración

Ejemplo de IaaS:



## PaaS : Platform as a Service

Es otro modelo en la nube que se encarga de dar acceso a una plataforma completa para desarrollar, ejecutar y administrar aplicaciones. Esto permite que los desarrolladores no se preocupen por la infraestructura subyacente.

Ejemplos de PaaS:



## SaaS : Software as a Service

Es el tercero de los modelos de distribución, en este caso de software y aplicaciones, este modelo se ha vuelto popular por ofrecer una alta conveniencia para el acceso, costes predecibles y otras facilidades

Ejemplos de SaaS



# **Tipos de Nubes**

## **Nubes Públicas**

Son aquellas infraestructuras que son externas a nuestra organización. Por decirlo de alguna forma, nosotros somos los administradores de la nube, pero no somos los dueños de los recursos de la misma. Quiere decir que todos los elementos físicos de hardware son propiedad de una entidad externa a nosotros, que por un módico precio u otro tipo de intercambio nos ceden esos medios físicos.

## **Nubes Privadas**

Para explicarlo siguiendo el hilo del punto anterior, en este caso nosotros somos los encargados de administrar esos recursos y, de forma añadida, somos los propietarios de esos recursos físicos. Por poner un ejemplo a muy pequeña escala, podríamos crear y asumir una cloud personal, o que se ubica en nuestro propio domicilio. Me explico. Podemos disponer de un ordenador retirado o que ya no usamos como el principal, que usamos como servidor de almacenamiento o incluso de aplicaciones, que sirven y nutren por ejemplo con aplicaciones a nuestro ordenador principal, como un servidor de almacenamiento al que podemos acceder desde la televisión como si fuese nuestro propio Netflix casero para ver películas o fotos que tengamos almacenadas, etc. Esto en resumen quiere decir que administramos tanto el hardware como los servicios que se proporcionan en esta Nube privada.

## Nubes Híbridas

Por temas de flexibilidad, escalabilidad o seguridad y acceso, nos encontraremos que en ocasiones preferimos optar por una solución híbrida. Habrá cierta necesidad de servicios o aplicaciones que queramos mantener bajo control total con infraestructuras locales. Luego aquellos servicios que puedan ser más susceptibles de cambios o modificaciones de recursos necesarios, es decir, que necesiten de adaptabilidad o escalabilidad, ya sea tanto de crecimiento o decrecimiento, son las que dejaremos en Clouds públicas, para no tener que abordar constantemente problemas con adquisición de nuevo hardware o equipamiento. La flexibilidad de este tipo de combinaciones, nos permiten ser mucho más concretos a la hora de resolver cualquier problema con nuestras necesidades.

## Ejemplo de Negocio

### Propuesta de Negocio

Desarrollar una plataforma web de **gestión de reservas online para pequeñas empresas** (restaurantes, peluquerías, eventos) que ofrezca una interfaz para cliente + panel de administración + recordatorios automáticos (correo / SMS). Monetización: suscripción mensual + comisiones por reserva extra.

### Soluciones comerciales en la nube – niveles

Vamos a elegir **IaaS** (Infraestructura como Servicio) y **PaaS** (Plataforma como Servicio) como los dos niveles, con ejemplos reales.

- IaaS – Amazon EC2(AWS)
- PaaS-Heroku

### Servicios que ofrecen más costes

#### Amazon EC2 (AWS) — IaaS

**Servicios ofrecidos** (para soportar nuestra app de reservas):

- Instancias virtuales con diferentes tamaños (CPU, RAM, tipos de red) para backend / servidores de aplicación. [Railsware+3GetDeploying+3Platzi+3](#)
- Almacenamiento persistente, balanceadores de carga, escalado automático, seguridad, redes, backups. [Documentación de AWS+2GetDeploying+2](#)

**Costes estimados:**

- Una instancia pequeña tipo t3.micro (≈1 GiB RAM, 1 vCPU) sale por unos **0.0104 \$/hora** lo que da cerca de **7-8 \$ al mes** si está siempre encendida. [Railsware+2aws-prep.com+2](#)

- Si se escala a algo más potente, por ejemplo 2 vCPU, 4-8 GiB, los costes suben aproximadamente a decenas o cientos de dólares al mes según configuración. [GetDeploying+1](#)

## Heroku — PaaS

### Servicios ofrecidos:

- Ejecución de aplicaciones en “dynos” (contenedores ligeros) donde Heroku se ocupa de la infraestructura, despliegue fácil vía Git, escalado más sencillo. [govery.com+2Back4App Blog+2](#)
- Bases de datos gestionadas (Heroku Postgres), cache (Redis), add-ons varios (monitorización, correo, integraciones), soporte de SSL, dominios personalizados. [Back4App Blog+2govery.com+2](#)

### Costes estimados:

- Dynos básicos (p.ej. *Standard 1x* o *Standard 2x*)  $\approx$  **25-50 \$/mes** dependiendo RAM y características. [Back4App Blog+1](#)
- Dynos de rendimiento  $\approx$  **250-500 \$/mes** en los niveles más altos si hay gran tráfico/computación. [govery.com+2Back4App Blog+2](#)
- Bases de datos gestionadas: Heroku Postgres básico/intermedio dependiendo del tamaño. Desde unos pocos dólares/mes para versiones pequeñas, hasta cientos para versiones premium. [Back4App Blog+1](#)

## Ventajas y desventajas de cada solución

### AWS:Ventajas

1. Muchísima flexibilidad: eliges sistema operativo, tamaño, red, etc.
2. Potencial de optimizar costes: usar instancias reservadas, apagar instancias cuando no se usan, autoescalado.
3. Escalabilidad y control fino: puedes montar arquitectura compleja (balanceadores, múltiples instancias, regiones).



### *AWS:Desventajas*

1. Gestión operativa más alta: tú eres responsable de mantener, parchear, configurar, monitorizar, etc.
2. Curva de aprendizaje: se requiere conocimiento DevOps/infra.
3. Costes escondidos: tráfico de red, backups, almacenamiento, licencias, etc. pueden aumentar el coste si no se controlan.

### *Heroku : Ventajas*

1. Muy buena experiencia de desarrollador: despliegue rápido, poca infraestructura que configurar.
2. Menor mantenimiento operativa: actualizar, seguridad, escalabilidad básicos ya gestionados.
3. Muchas integraciones (add-ons) y servicios auxiliares listos para usar.

## Justificación de elección

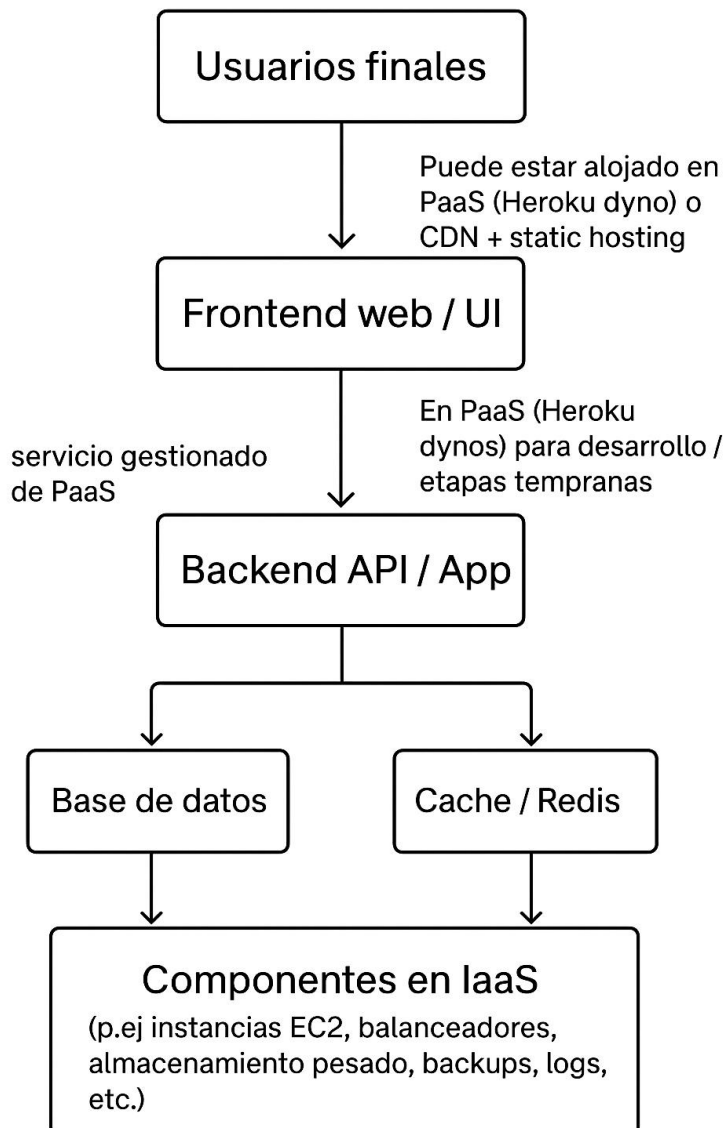
Para la plataforma de reservas:

En etapas iniciales Heroku ofrece rapidez de puesta en marcha, ideal para probar una idea, con costes fijos que se pueden calcular de forma sencilla.

Cuando la plataforma crezca, tenga mucho tráfico o necesite ser optimizada, migrar de forma parcial o completa hacia AWS EC2 permitirá mayor control y menor coste total, siempre dependiendo de como de bien se gestionen los recursos.

## Diagrama de interacción entre los niveles de servicio

Siguiendo lo indicado en los puntos anteriores hemos realizado un diagrama en el que hemos intentado reflejar cómo interactúan los componentes usando estos niveles:



En las etapas iniciales de este proyecto podríamos hacer que todo funcione sirviendonos de las diferentes tecnologías disponibles teniendo en cuenta diferentes factores:

Cuando la demanda crece o la exigencia del servicio así lo requiere se puede migrar a otros servicios más amplios (IaaS) como EC2.

Cuando el almacenaje en la nube se hace demasiado grande estos archivos o elementos almacenado podrían ir a S3, que es un servicio gestionado de AWS, que es accesible tanto desde sistemas PaaS como IaaS.