

[Home](#)[About Us](#)[Solutions](#)[Industries](#)[Contact](#)

# IDENTIFICANDO COMPONENTES Y MODELOS DE SERVICIO



- PEDRO ARZA
- ALEJANDRO SAINZ
- JUAN DAVID SERVELLÓN

[Home](#)[About Us](#)[Solutions](#)[Industries](#)[Contact](#)

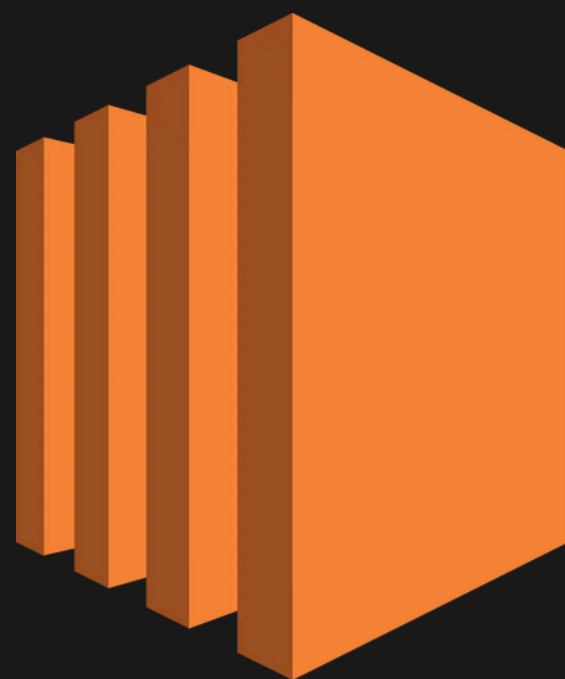
# PROPUESTA DE NEGOCIO

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.

[Home](#)[About Us](#)[Solutions](#)[Industries](#)[Contact](#)

# SOLUCIONES COMERCIALES EN LA NUBE

Hemos elegido IaaS (Infraestructura como Servicio) y PaaS (Plataforma como Servicio) como los dos niveles, con ejemplos reales.



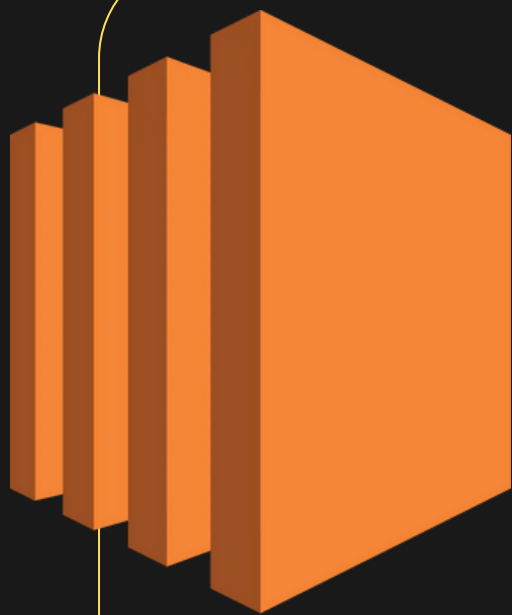
Amazon EC2 (AWS)



Heroku



# SERVICIOS Y 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.
- Almacenamiento persistente (volúmenes EBS), balanceadores de carga, escalado automático (Auto-Scaling), seguridad (VPC, IAM), redes, backups.

### Costes estimados:

- Una instancia pequeña tipo t3.micro ( $\approx$  1 GiB RAM, 1 vCPU) sale por unos **US\$0.0104/hora** lo que da cerca de **US\$7-8 al mes** si está siempre encendida.

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.



# SERVICIOS Y COSTES



heroku

## Heroku - PaaS

### Servicios ofrecidos:

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

### Costes estimados:

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



# VENTAJAS Y DESVENTAJAS DE CADA SOLUCIÓN

## AWS EC2 (IaaS)

### Ventajas

- Flexibilidad: elección de sistema operativo, tamaño y red.
- Optimización de costes: instancias reservadas, apagado y autoescalado.
- Escalabilidad y control: permite arquitecturas complejas con balanceadores, múltiples instancias y regiones.

### Desventajas

- Mayor gestión operativa: responsabilidad de mantenimiento, parches y monitoreo.
- Curva de aprendizaje: requiere conocimientos de DevOps e infraestructura.
- Costes ocultos: red, backups, almacenamiento y licencias pueden encarecer el servicio.

## Heroku (PaaS)

### 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.

### Desventajas

- Costes más altos conforme escalas: los dynos de rendimiento y bases de datos grandes tienen precios elevados.
- 2. Menos control: limitaciones en configuraciones muy específicas, hardware, redes, etc.
- 3. Posibles limitaciones de rendimiento para tareas muy intensivas o tráfico altísimo.

[Home](#)[About Us](#)[Solutions](#)[Industries](#)[Contact](#)

# JUSTIFICACIÓN

Para la plataforma de reservas suggested:

- En etapas iniciales / MVP, Heroku ofrece rapidez de puesta en marcha, menos fricción, ideal para probar idea, con costes previsibles (ventaja de tiempo / foco en producto, no infraestructura).
- Cuando la plataforma crezca, tenga mucho tráfico o necesite optimizar costes/performance, migrar parte o todo a AWS EC2 permitirá mayor control y posiblemente menor coste total si gestionas bien los recursos.



# DIAGRAMA

En etapas iniciales, todo podría correr en PaaS: frontend, backend, base de datos, cache.

- Cuando la demanda crece, partes pesadas (logs, backups, tareas batch, procesamiento intensivo) se pueden mover a instancias IaaS como EC2.
- Cloud storage pesado o archivos estáticos podrían ir a S3 (servicio gestionado de AWS) accesible tanto desde PaaS como IaaS.

