# Plan de Sistemas

Se pretende definir la arquitectura necesaria para que la aplicación EventPlanner pueda prestar su servicio web a cualquier usuario.

Para ello, se toman dos puntos de vista distintos para estructurar dicho despliegue:

* **Amazon Web Services** (AWS): por medio de la contratación de una serie de servicios en línea que conformen una Cloud privada.
* Un entorno físico, como pueda ser un laboratorio o una empresa, conformando un **Centro de Procesamiento de Datos** (CPD).

Esto es, suponiendo que somos una pequeña empresa, buscamos la mejor solución para poder desplegar nuestra aplicación, comparando un servicio en la nube con un servicio local.

Para empezar, se sientan las bases de la arquitectura lógico y física a definir por medio de la arquitectura ya presentada al comienzo del proyecto, así como por los requisitos de despliegue explicados en el apartado anterior.

Además, se tiene en cuenta que, debido al presupuesto que una pequeña empresa de reciente formación pueda manejar, se deben ajustar los precios de ambas posibilidades lo máximo posible. Es decir, conseguir equipamiento lo más barato posible, pero sin descuidar los requerimientos mínimos establecidos, para asegurar la alta disponibilidad y la escalabilidad (sin olvidar la persistencia de los datos).

Tras un análisis exhaustivo de los componentes necesarios, se presentan los siguientes presupuestos:

## AWS

Mostramos los componentes elegidos para conformar la arquitectura en AWS, destacando de cada uno sólo aquellas características por las que nos hemos decantado por cada uno de ellos:

Comenzando por el punto de vista del usuario, es necesario contar con:

* Route 53: servicio web DNS escalable y de alta disponibilidad en la nube.
  + Conecta efectivamente las solicitudes de los usuarios con la infraestructura desplegada en AWS, sean cuales sean sus tipos de instancias.

BD: Varias EC2 (volátil) en zonas distintas para mayor disponibilidad y apuntando a un mismo volumen -> Amazon S3 (persistente).

API: varias instancias en EC2 distintas y balanceador de carga [Elastic Load Balancing] para repartir las peticiones.

* Mínimo 2 en zonas distintas.

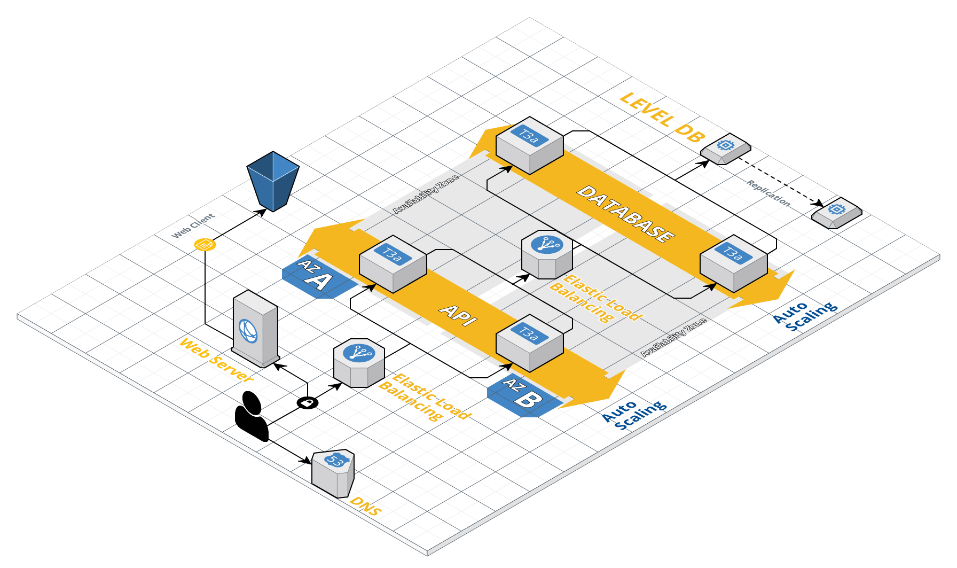


Ilustración Arquitectura en AWS

Cliente web desde una EC2.

S3 es el DNS

T3.a.: por ahorro en coste y porque tiene un uso general, optimizado para un rendimiento medio y picos eventuales de carga.

EBS: almacenar de forma persistente los datos de la BD y realizar una réplica.

Route 53: DNS.

CloudFront: sirve el cliente web.

S3: almacenar HTML, CSS y JavaScript dinámico para servir la página.

Zonas diferenciadas: en la región Irlanda EU-WEST-1 se pueden conseguir hasta 3 zonas distintas, dispondremos de 2.