

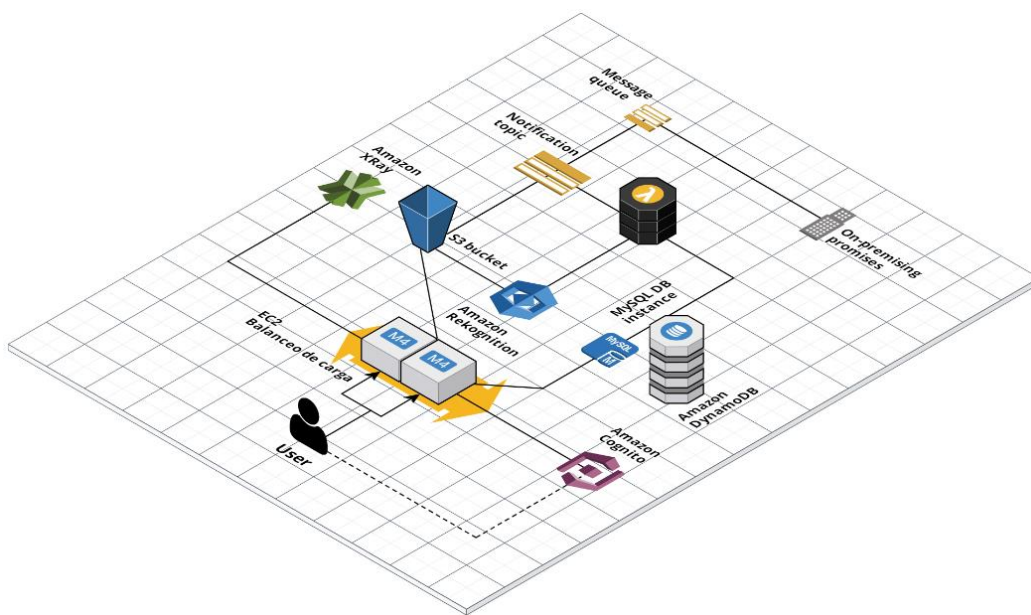
## Introducción

Se va a realizar una aplicación con la cual poder trackear una escena en la que un objetivo principal es reconocer y gestionar rostros de personas que discurren por la misma.

Para ello es necesario diseñar una arquitectura que, de manera simple y eficiente, gestione las imágenes obtenidas. Puesto que Amazon Web Services (AWS en adelante) nos permite cubrir toda la casuística pertinente, se va a basar todo el diseño de la arquitectura en estos servicios. Además como la única restricción es la de realizar dicho proyecto empleando máquinas virtuales (VM en adelante), se utilizará EC2 de AWS por estar basado en instancias de VMs.

## Diseño

El diseño que hemos considerado para nuestra arquitectura es el descrito a continuación



### Componentes:

- Amazon Cognito: Se encarga de la gestión de los diferentes usuarios que tendrán acceso al sistema. Su función principal es la gestión de identidades y autenticación en la nube de forma simple y segura
- EC2: Proporciona una amplia selección de tipos de instancias optimizados para adaptarse a diferentes casos de uso. En este componente se crearán las VMs que nos permitirán gestionar la información que será procesada por el sistema
- X-Ray: Consiste en un depurador de aplicaciones distribuidas. Su principal uso es en aplicaciones que utilizan microservicios o arquitecturas serverless
- S3 Bucket: Almacenamiento de los objetos multimedia que tratará el sistema
- Rekognition: Encargado de procesar un streaming de video o una serie de imágenes. Una de las funcionalidades principales es la detección de caras, siendo necesario para el sistema que se va a desarrollar
- Lambda: Implementación de la gran parte de la lógica del sistema, obteniendo los resultados procesados por Rekognition y lo almacena en la BBDD
- MYSQL DB o DynamoDB. Son dos alternativas para almacenar los resultados obtenidos del servicio AWS Rekognition