

## Práctica 3 AWS

El siguiente ejercicio está pensado para poner en práctica los conocimientos de AWS Lambda/EC2/S3 adquiridos hasta el momento. El alumno deberá implementar un programa usando los servicios de amazon, que simule una aplicación tipo red social/Twitter. Se deberá implementar una base de datos, funciones lambda necesarias y páginas web que den los servicios especificados a continuación. El ejercicio está dividido en varias partes, cada una de ellas cubrirá una de las herramientas vistas.

Notas a tener en cuenta:

- No se ha usado RDS por su alto coste, pero se deja libertad al alumno el hecho de usarlo e implementar las bases de datos con esta herramienta
- Las bases de datos usadas en los ejemplos de clase estaban basadas en mysql, y alojadas dentro de ordenadores EC2. Se deja libertad para cambiar el tipo de base de datos y alojamiento siempre que sea una máquina con acceso en internet.

Enunciado:

Se pide implementar una página web que permita publicar comentarios con archivos multimedia adjuntos (videos o imágenes), gestionar usuarios y propiedad de sus comentarios/archivos, listar comentarios propios y de otros usuarios, crear listas de usuarios seguidores/seguídos, realizar respuestas a mensajes de otros usuarios. Las funcionalidades principales serán las siguientes:

- Login y registro de usuarios
- Por cada usuario, acceso a una página web que permita crear comentarios y subir ficheros de video/imágenes adjuntas
- Listado de comentarios propios y reproducción de adjuntos

Partes opcionales:

- Gestión de usuarios seguidos:
  - Opción de búsqueda de usuarios
  - Opción de añadir un usuario a una lista de “seguidos”
  - Mostrar mensajes de usuarios seguidos en una página web, ordenados por fecha de publicación
- Gestión de respuestas a mensajes
  - Votación positiva/negativa (like/dislike)
  - Respuesta a un mensaje
- Seguridad en accesos a páginas (se deja al alumno implementar sistemas tipo “cookies”, sesiones, passwords, videos/posts privados privados....)

## Sección1: Gestión de usuarios y contenido (5 puntos, obligatoria)

### - Parte 1: Diseño de la base de datos

Se deberá diseñar e implementar una base de datos que permita almacenar la siguiente información:

Usuarios: ID/Nombre de usuario, correo electrónico, contraseña, frase de recuperación de contraseña, avatares, biografía, etc...

Mensajes: Identificador de mensaje, ID Usuario que lo creó, mensaje, dato adjunto (solo uno), tipo de dato (video/imagen)

Opcional: Tablas y datos para secciones de gestión de seguidores, votaciones (like/dislike) y respuestas. El alumno tiene libertad para añadir tablas y otras características a la base de datos para implementar nuevas funcionalidades.

### - Parte 2: Sistema de login y registro

Se puede reutilizar código de prácticas anteriores y adaptarlo a esta base de datos. Se deberá implementar una página web para acceso al sistema y una función lambda que la gestione. El sistema funcionará de la siguiente manera:

- Pide nombre de usuario y contraseña
- Se validan en la función lambda.
  - Caso correcto: Redirige a la página personal del usuario
  - Caso incorrecto: Se tienen 3 intentos para entrar correctamente. En caso de no conseguirlo, la cuenta quedará invalidada.
- EXTRAS: (+1 punto adicional si se realiza, nota máxima de 6 en esta sección)
- Opciones de altas de usuario:
  - Se pueden crear nuevos usuarios en el sistema. Para ello se pedirán todos los datos necesarios para identificar al usuario, y se almacenará en la base de datos
- Opciones de recuperación de contraseñas perdidas:
  - Si la cuenta no ha sido deshabilitada, se pedirá la frase de recuperación de contraseña para validarla
- Cualquier añadido que se crea necesario para mejorar el uso del sistema:
  - Avatares, biografía, etc...

### - Parte 3: Sistema de gestión de comentarios (reutilizar ejemplos de clase)

Por cada usuario, se pueden subir un único archivo adjunto al mensaje que quiera publicar. Cada vez que se cree un mensaje, habrá que actualizar la base de datos con (al menos) la información de ese mensaje, archivo adjunto, usuario que lo creó, datos de tipo de archivo adjunto. La forma de ejecución podría ser la siguiente:

- Desde la página personal del usuario, se puede acceder a la sección de publicación de mensajes, con una opción a archivos adjuntos(botón, menú, como se desee implementar)
- Se le presenta una página web con el formulario para introducir el mensaje y el archivo adjunto. Una vez escrito el mensaje y accionado el botón de publicar, se enviarán los datos por separado:
  - Datos del mensaje a una función lambda que actualizará la base de datos
  - Archivo adjunto al bucket s3
  - OJO: No se pueden repetir archivos con el mismo nombre en el bucket. Dos opciones:
    - Carpetas para cada usuario privadas
    - Crear nombres “random” para cada archivo adjunto, usando por ejemplo el nombre de usuario como prefijo.

Una vez creado el mensaje, el usuario podrá listar esos mensajes desde otra página web:

- Ver todos los mensajes publicados hasta ese momento

## Sección 2: Gestión de seguidores, sus mensajes, respuestas, votos y seguridad

Esta sección es opcional, y se deja libertad al alumno para poder implementarla.

### - Parte 1: Búsquedas y gestión de usuarios seguidos (2 puntos)

Hasta este momento, se debe de tener un sistema de subida de ficheros a S3, gestión de base de datos, y funciones lambda que permitan realizar búsquedas sobre mensajes propios del usuario. La siguiente parte permitirá buscar y reproducir mensajes de cualquier usuario que haya en el sistema. Para ello, se pide lo siguiente:

- Una vez ha realizado login el usuario, podrá acceder a su página personal para publicar mensajes, o acceder a la sección “comunidad”, donde podrá buscar y reproducir los mensajes del resto de usuarios. Añadir el acceso a esa nueva sección a la práctica.

- Sección comunidad:
  - Contendrá una caja de texto donde el usuario podrá escribir palabras de búsqueda. Esas palabras deberían coincidir con los nombres con los que se registraron otros usuarios (se deja libertad en su implementación). Junto con esa caja de texto, habrá un botón que permitirá el envío de esa búsqueda a una función lambda.
  - La función lambda que reciba la búsqueda accederá a la base de datos, y devolverá el resultado de esa búsqueda. En concreto, debe devolver usuarios que coinciden con los patrones de búsqueda (se deja libertad en implementar “similitudes”).
  - Como extras, se podría devolver también un acceso a su “perfil”, donde se pueden comprobar los últimos mensajes del usuario buscado
  - Una vez recibidos se ofrece al usuario la opción de “seguir” a esos usuarios.
  - EXTRAS:
    - Cualquier tipo de gestión de usuarios seguidos:
      - Seguir, listado de seguidores, silenciar, dejar de seguir, etc...

## - Parte 2 (opcional): Gestión de mensajes de usuarios seguidos (2 puntos)

En la página de “comunidad” comentada anteriormente, se mostrarán los mensajes más recientes de todos los usuarios seguidos, ordenados por fecha de publicación:

- Se puede añadir una columna “fecha de publicación” a la tabla que almacene mensajes.

En los mensajes mostrados, igualmente se permite algún tipo de interacción:

- Likes/dislikes, respuestas, menciones, ocultar...

Creación de hilos de conversación. Para simplificar, se recomienda permitir únicamente una respuesta a cada mensaje. Podéis tener dos tipos de mensajes:

- Mensaje original: El primero de un hilo. Puede tener un identificador a un mensaje “hijo”, que sería una respuesta.
- Mensajes respuesta: Deben tener obligatoriamente un identificador a un mensaje “padre”, y opcionalmente un identificador a un mensaje “hijo” que sería una nueva respuesta

En el caso de tener hilos de mensajes, se deja libertad para poderlos mostrar al usuario. Se aconseja mostrarlo de una de las dos siguientes maneras:

- Opción 1: Mostrar todo el hilo completo junto con el resto de mensajes
- Opción 2; Si un mensaje era el inicio de un hilo, añadir un botón “mostrar hilo” que despliegue esa conversación

- Parte 3 (opcional) : Seguridad de acceso a funciones lambda y áreas privadas de usuario (1 punto)

Se deja libertad al alumno para implementar algún sistema de validación para poder acceder a las funcionalidades del sistema y datos de usuario de forma segura. Se puede usar un sistema de cookies, tokens de sesión, mensajes cifrados, etc.... cualquier tipo de seguridad añadida será tenida en cuenta.

## EVALUACIÓN

Para poder considerar la práctica aprobada (nota=5), se pide realizar las siguientes partes de forma obligatoria:

- Sistema de login (validación por usuario/contraseña)
- Sistema de publicación de mensajes y subida de archivos adjuntos (visto en clase)
- Sistema de reproducción de mensajes publicados (visto en clase)

Cada una de las partes opcionales podrán sumar puntos para llegar al 10. Su valoración será la siguiente:

- Gestión de creación de usuarios y recuperación de contraseñas: 1 punto
- Gestión de búsquedas de usuarios, gestión de seguidores y mostrar su perfil/mensajes: 2 puntos
- Gestión de “comunidad”, con mensajes de usuarios seguidos, votos, respuestas, hilos, etc...: 2 puntos
- Seguridad en accesos a páginas: 1 punto