

RedES: Conexiones

Siguiendo el desarrollo de la red social se procederá con la fase de diseño e implementación de sistema de conexiones entre usuarios. Esta fase será desarrollada íntegramente haciendo uso de la base de datos Neo4j.

Este sistema debe ser capaz de gestionar todas las conexiones existentes entre los usuarios de la red social. Se ha de tener en cuenta que las conexiones entre usuarios podrán ser de diversa tipología: familia, amistad, académico o laboral; pudiendo tener una o varios tipos de conexión al mismo tiempo.

Existirán unos usuarios con características especiales los cuales se corresponden con los centros educativos en los que los usuarios estudian o han estudiado y empresas en los que los usuarios trabajan o han trabajado.

Además, este sistema deberá ser capaz de gestionar los mensajes enviados entre los distintos usuarios. Se deberá almacenar el id de la conversación, el número de secuencia del mensaje, la fecha de envío y el mensaje en si. Por razones de simplicidad, los mensajes solo se podrán enviar de un usuario a otro usuario.

Por último, el sistema deberá ser capaz de gestionar publicaciones. Una publicación es producida por un usuario y consta de título, cuerpo de la publicación y fecha de la publicación. Además, se deberá gestionar las menciones de las publicaciones. Una publicación podrá mencionar a uno o varios usuarios.

El sistema permitirá buscar nuevas conexiones entre usuarios a partir de toda la información almacenada en esta base de datos.

Requisitos de la práctica

Diseñar e implementar una api (únicamente las funciones) que gestione la funcionalidad básica de la plataforma, así como el fichero/consultas que permita/n inicializar la base de datos con la información necesaria para probar toda la funcionalidad exigida.

Usuarios

- Gestión de usuarios y conexiones.
- Consultas:
 - Obtener los amigos y familiares de un usuario determinado.
 - Obtener los familiares de los familiares de un usuario determinado.

Mensajes

- Gestión de mensajes e información asociada.
- Consultas:
 - Obtener todos los mensajes enviados de un usuario determinado a otro usuario determinado después de una fecha especificada.
 - Obtener la conversación completa entre dos usuarios determinados.

Publicaciones

- Gestión de publicaciones e información asociada.
- Consultas:
 - Obtener todos los usuarios mencionados por un usuario determinado los cuales tengan una relación laboral con el usuario que los mencionó.

Búsqueda de nuevas conexiones

- Consultas:
 - Obtener los usuarios (terceros) que, no teniendo relación con un usuario determinado (primero), tengan alguna relación en uno o varios saltos de relación con los usuarios (segundos) que tienen relación con el usuario determinado. Se podrá definir el número de saltos máximo en la consulta. En la consulta se mostrará el usuario segundo del que parte la relación con los terceros y el número de saltos de relación entre dichos usuarios y se ordenará por número de saltos.
 - Obtener los usuarios (terceros) que, no teniendo relación con un usuario determinado(primero), tengan alguna relación con los usuarios (segundos) que tienen relación con el usuario determinado. Solo se mostrarán las relaciones entre usuarios que tengan más de un número especificado de mensajes. Ordenar el resultado primero por el número de mensajes entre el primer usuario y los segundos y después por el número de mensajes entre segundos y los terceros.

Normativa de realización, entrega y evaluación de la práctica:

- La práctica se realizará y entregará en grupos de hasta dos integrantes.
- La práctica se realizará en python y haciendo uso de Neo4j.
- La práctica deberá ir acompañada de las pruebas necesarias para comprobar el buen funcionamiento de la funcionalidad que se pide.
- La entrega se compondrá de un único fichero ZIP, que contendrá el directorio del proyecto con un listado de las dependencias necesarias.
- Se considerará suspensa toda práctica cuyo fichero comprimido no contenga los ficheros fuente.
- La entrega deberá hacerse mediante el campus virtual antes del domingo 30 de noviembre de 2021 a las 23:59 horas (hora peninsular en España).
- Las prácticas entregadas fuera de plazo, serán calificadas sobre 9. Por cada día de retraso en la entrega se reducirá el rango de calificación en 0,2 puntos.
- La entrega se compondrá de un único fichero .py renombrado con el número de la práctica P#, seguido del número del grupo G#, y finalmente seguido con el nombre y el primer apellido de los alumnos integrantes del grupo, separados mediante guiones bajos '_'.
Ejemplo: P3_G25_Quijote_de_la_Mancha-Sancho_Panza.zip

- Cualquier sospecha de COPIA entre dos o más prácticas o de código obtenido en internet derivará en la calificación de 0 para todos los alumnos involucrados en la evaluación en curso y la siguiente. En caso de que el alumno tenga duda de que el código pueda ser susceptible de ser entendido como copia, consultar con el profesor antes de la entrega.