



Tarea 2 BD : MySQL - PHP

USMwer

Gabriela Acuña, rol : 201973504-7

1. Descripción breve del problema, incluyendo los supuestos que realizó para el trabajo.

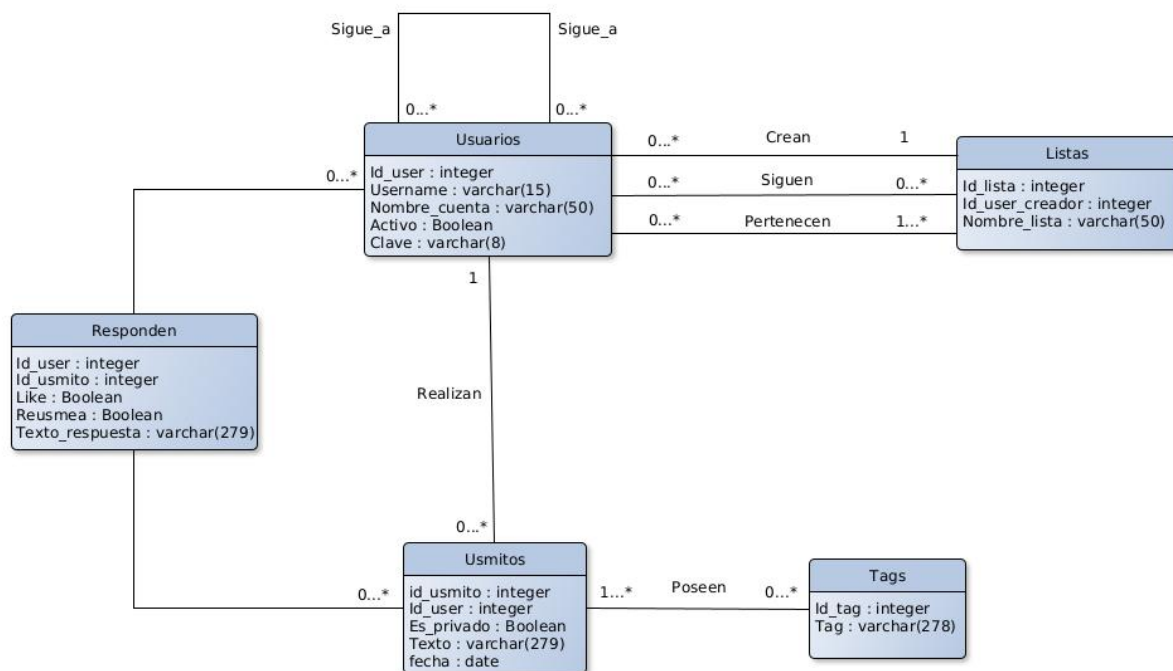
El problema a resolver es una red social en base a textos cortos, usmitos, y usuarios que pueden interactuar de diferentes formas con estos textos y entre sí. Respecto a los usmitos, un usuario puede crear uno, o si es uno creado por otro usuario puede, reusmearlo (compartirlo en su propio feed), responderlo (comentarlo, sin que sea mostrado en su propio feed) y/o darle me encanta. En esta implementación un usuario no podrá citar un usmito preexistente en un nuevo usmito. Además, el usuario puede escoger si publicar un usmito de forma pública o privada, para que solo sus amigos cercanos lo vean. Considerando a los amigos cercanos, los usuarios a los que siga y lo sigan de vuelta. Por último, un usmito puede contener uno o más tags, pero no es obligación, y para generar la tendencia de estos, se creará una vista que contenga los nombres de los tags y sus apariciones en usmitos tanto públicos como privados.

Respecto a los usuarios, estos pueden crear listas de otros usuarios para poder ver los usmitos publicados específicamente por ellos. Estas listas son públicas y pueden ser seguidas por otros usuarios. Si un usuario reusmea un usmito, éste no contará como que fue escrito por él, por lo que no sumará a las estadísticas presentadas en la página de su perfil.

Por otro lado, para todas las especificaciones de largos de textos se tomó como referencia la red social en la que está basada USMwer.

Para el desarrollo de esta tarea se utilizará LAMP con mariaDB y DBeaver, desde Manjaro Arch Linux. Los triggers y views se agregaran a lo largo del desarrollo de la implementación, pero a grandes rasgos, como ya fue mencionado, se creará un view para ver las tendencias de los tags y las que se estimen convenientes para presentar los diferentes datos necesarios en la aplicación web. Sobre triggers, se utilizarán para actualizar diferentes datos en las tablas que sea necesario y específicamente para el atributo Mutuo de la tabla User1_sigue_User2, ya que si un usuario sigue a otro, cuando se intente ingresar la relación inversa, simplemente se cambia a True el atributo mutuo.

2. Modelo de datos conceptual, expresado en un diagrama de clases UML, utilizando una herramienta CASE, como starUML, yEd Graph Editor o Draw.io.



3. Modelo de datos relacional obtenido utilizando el proceso de 8 pasos vistos en clases. También expresado en un diagrama de clases UML.

Los tipos de datos Boolean serán implementados como INTEGER(1) en MySQL, considerando al 1 True y 0 False. Las entidades asociativas se utilizaron como entidades intermedias de relaciones, ya que todas pertenecían a relaciones M:N y cumplían con la regla 5.

