

## Revisión del diseño de las bases de datos (a fecha de 16/12/12)

### Base de datos de la web

- En lugar de guardar tipos de usuario, creo que es mejor crear grupos de usuarios. En principio, tendríamos tres: alumnos, profesores y administradores. Las ventajas de los grupos de usuarios son básicamente dos:
  - Es un mecanismo muy conocido para restringir las acciones que los usuarios pueden realizar. En lugar de inventarnos nuestro propio mecanismo, si usamos un mecanismo estándar como este el funcionamiento del proyecto será mucho más intuitivo.
  - Las comprobaciones se simplifican. Los grupos de usuario son aditivos: es decir, los profesores pertenecen a los grupos profesores y alumnos, y los administradores pertenecen a los tres. Para arrancar una máquina virtual, sólo habrá que preguntar si el usuario pertenece al grupo alumnos. Para editarla, sólo habrá que preguntar si el usuario pertenece al grupo profesores, y así sucesivamente.
- El curso de la asignatura nos fastidia en el caso de cualquier asignatura: en segundo puedes cogerte asignaturas de cuarto si quieres. Además, es redundante: cada asignatura tiene un identificador único, que es suficiente para identificarla. Lo que sí que falta es el curso académico: el identificador único y el año académico son los que identifican de forma única un curso de una asignatura.
- Estás asumiendo que cada grupo tiene asociada una única máquina virtual. Esto no es verdad: puede tener asignadas de 0 a  $n$  máquinas virtuales.
- No uses enteros para los identificadores únicos: usa enteros largos para tener las espaldas bien cubiertas.
- Las contraseñas deberían guardarse cifradas. Además, yo permitiría contraseñas de hasta 32 caracteres. En cualquier caso, la clave de cifrado estará metida “a pelo” en el código fuente para poderlas descifrar.
- Para identificar los tipos de usuario, es mejor tener otra tabla: tarde o temprano, va a terminar así por la normalización. Yo metería una tabla GrupoDe, que asocia a cada identificador de usuario un (en el caso de tu diseño) o uno o más (en el caso de que haya grupos de usuario) identificadores de grupo (i.e. Alumnos, Profesores o Administradores).
- 13 caracteres para el identificador de grupo... se me hacen pocos. Yo pondría 20 como mínimo.
- Yo llamaría a GrupoDeUsuarios GruposDeClase para que todo sea mucho más claro.
- En Grupo, estás asumiendo que sólo existe una MV asociada a cada grupo. Puede haber entre 0 y  $n$ . Además, la asociación entre grupos y máquinas debería estar en una tabla independiente para que esto sea posible.

Por otra parte, lo de guardar la letra no me termina de convencer. Ahora estoy muy espeso como para proponerte una solución, pero pensaré en esto más tarde.

## Base de datos del servidor principal

- En principio, la base de datos de la web y la del servidor principal tiene que ser la misma: si no es así, hay que implementar un mecanismo que vuelque los cambios en las máquinas virtuales a la base de datos de la web. **De todas formas, separar las dos bases de datos me parece razonable.**
- Lo mismo con las claves: mete enteros largos para cubrarnos bien las espaldas.
- No hay dirección del socket y dirección del puerto: lo único que hace falta para conectarnos a un servidor de máquinas virtuales es la IP y el puerto. El socket se crea y se destruye dinámicamente para realizar las comunicaciones.
- Creo que no estaría mal que las imágenes tuvieran una descripción asociada (básicamente, para que los usuarios puedan distinguirlas bien).

## Base de datos del servidor de máquinas virtuales

- 50 caracteres para una ruta absoluta se quedan cortos sí o sí. Para no alargar este campo un montón, debemos guardar rutas relativas (es decir, en lugar de guardar /home/luis/images/802354/Windows7 guardar 802354/Windows7).
- ¿Por qué no usas el puerto para identificar cada una de las máquinas activas? Cada una tendrá su propio puerto en el servidor VNC: no los pueden compartir.
- Para las direcciones MAC, hay que guardar sólo 12 caracteres. Para ahorrar espacio, puede ser interesante guardar sólo los 6 bytes (a cada byte le corresponden 2 dígitos hexadecimales; por eso hay 12 caracteres que guardar).
- Para la contraseña, usa algo grande: 64 caracteres. Con esto, las contraseñas serán ya muy largas.
- No veo por qué hay que separar MAC, puerto y contraseña en otra tabla. Si lo tenemos todo junto en esta, no habrá redundancia y el acceso a los datos será más rápido.

## Cosas que faltan en el diseño

- **Auto incremento para todas las claves: así, será MySQL el que genere las claves y no nosotros.**
- Definir nuevos usuarios en MySQL. No es seguro utilizar el usuario root todo el rato. Es necesario crear como mínimo un usuario en cada una de las tres bases de datos.
- Definir los privilegios de los profesores, alumnos y administradores en la base de datos de la web. No todos podrán usar todas las tablas para lo mismo, y esto hay que definirlo ahora porque después hay que implementarlo.