

Comentarios acerca del modelo (versión 1.3 – 06-06-2024)

1. En el modelo final, no se contemplan los pagos entre miembros, sino que la aplicación una vez que “liquida” saldos guarda el estado en la tabla “**payments**” y envía un correo indicando lo que cada uno debe al grupo (o bien lo que le deben a él si el saldo es positivo).
2. Por supuesto, se registran todos los pagos (tabla **expenses**) que se realizan con algún concepto (**description**) y que tienen asociado un usuario pagador (**payer_user_id**).
3. **IMPORTANTE:** Las invitaciones sólo se pueden realizar sobre usuarios registrados, para simplificar el sistema. Por ello, se ha incluido también el “**username**” que pueda facilitar la identificación y búsqueda de usuarios a invitar.

Vamos a los detalles tabla a tabla:

users

mail y **username** serán campos con UK (unique key) y NN (no nullables). También el campo **password** será NN.

El resto de campos son opcionales (en base de datos),

El campo “**active**” se usa simplemente para el borrado lógico del usuario (deshabilita las opciones de acceso para el usuario).

invitations

Como sólo se puede invitar a usuarios registrados, independientemente del campo de búsqueda que se utilice, aquí guardamos el id (se relaciona con la tabla usuarios a través del campo **user_id**).

Dispone de dos flags (campos booleanos), que son:

- **accepted:** Indica si se ha aceptado (valor a 1) o no (valor a 0) la invitación. Por defecto 0.
- **active:** Indica si se ha “rechazado” (se pondrá el valor a 0) y que hace que no esté ni “pendiente” (**accepted** = 0) ni “aceptada” (**accepted**=1). Por defecto este valor está a 1 (si el campo “**accepted**” está a 0, significa que está “pendiente”). Por tanto, los posibles estados son:

Campo accepted	Campo active	Estado
0	1	Pendiente (por defecto)
1	1	Aceptada
0	0	Rechazada

expenses

Es la tabla de gastos se registra, junto con los datos del gasto (**description**, **amount**, **date**) el grupo al que pertenece (**group_id**) y el usuario pagador (**payer_user_id**).

El campo “**active**” es importante en nuestra lógica, porque será 1 por defecto cuando se crea el gasto y significa que no está liquidado. Cuando desde la aplicación se ordene

saldar el grupo, realmente la aplicación lo que hará será saldar los gastos asociados, es decir, poner a 0 este valor (similar a un borrado lógico), y volcar a la tabla correspondiente (**payments**, en este caso) la información de saldo de cada miembro.

payments

Es la tabla de pagos (resultado de saldar un grupo).

Es muy intuitiva y simplemente refleja el usuario, el grupo, y el saldo (si es negativo es importe que se debe al grupo, si es positivo se lo deben al miembro). También se guarda la fecha. Esta info es enviada por correo al usuario, y sólo se ve en la APP cuando es “provisional”, es decir, cuando los saldos son calculados (momento en el que aún no están volcado aquí esos datos)

groups

Se guarda junto con la descripción y la categoría (en base a un listado preestablecido que tendremos en BD) el usuario creador (**creator_user_id**, que es FK de **users**), y por tanto usuario administrador.

group_categories

Es la tabla maestra que contiene la relación de categoría que puede tener un grupo.

group_members

Es la tabla que relaciona los usuarios con el grupo. Por tanto, contiene los miembros de un grupo.

Se guarda aquí el % que se asigna a cada usuario. Por defecto debe ser equitativo para un grupo, pero al dar la posibilidad de que uno o varios tengan un valor distinto, debemos contemplarlo en la BD para que no se recalcule. Por eso existe un campo **equitable** (por defecto 1), si se establece un % manual, este valor pasará a ser 0 (y volvería tener 1, si se vuelve a marcar como “equitativo” en la aplicación). El campo percent realmente guarda un valor entre 0 y 1 (multiplicado por 100 sería el % en el que participa en los gastos totales del grupo).

messages

Por último, la tabla de mensajes. Es muy intuitiva. Guarda el timestamp, junto con el usuario y grupo al que pertenece el mensaje, que se graba en el campo “**message**”, de tipo TEXT para que pueda albergar cadenas superiores a 255 caracteres (máximo del varchar).