

## Comentarios acerca del modelo (versión 1.1 – 14-05-2024)

1. En el modelo final, no se contemplan los pagos entre miembros, es decir, sólo se registran los pagos (**expenses**) que se realizan con algún concepto (**description**) y que tienen asociado un usuario pagador (**payer\_user\_id**).
2. IMPORTANTE: Las invitaciones sólo se pueden realizar sobre usuarios registrados, para simplificar el sistema (si sobra tiempo, se estudiará escalar esta funcionalidad). Por ello, se ha incluido también el "**username**" que pueda facilitar la identificación y búsqueda de usuarios a invitar.
3. Se han eliminado las tablas de estado, se utilizarán campos booleanos tipo active, equitable, accepted... etc.

### Vamos a los detalles tabla a tabla:

#### **users**

---

**mail** y **username** serán campos con UK (unique key) y NN (no nullables)

El resto de campos son opcionales (en base de datos), pero si se define como obligatorios en la funcionalidad, también deberíamos cambiarlo aquí.

El campo "**image**" guardaría url de la imagen en el servidor (o la clave que permite recuperar una imagen, según la librería que usemos), podemos decidir guardar directamente la imagen aquí en la tabla (podemos preguntarlo, pero creo que es más normativo guardarla fuera).

El campo "**active**" se usa simplemente para el borrado lógico del usuario (sólo si le damos la posibilidad al usuario de "darse de baja" en la aplicación se usaría).

#### **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" o "archivado" o el nombre que le pongamos (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")

## **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).

## **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. Si no os parece muy intuitivo podemos cambiar el nombre de ese campo.

## **group\_categories**

---

Es la tabla maestra que contiene la relación de categoría que puede tener un grupo. Dijimos que, dependiendo de este valor, cargaríamos una imagen (avatar, logo o similar) en la pantalla de creación de grupo (y podría usarse en los listados de grupos).

## **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)

## **messages**

---

Por último, la tabla de mensajes. Es muy intuitiva.

Como comentario, el campo “**message**” es de tipo TEXT para que pueda albergar cadenas superiores a 255 caracteres (máximo del varchar)