|  |  |
| --- | --- |
| 0down vote[favorite](http://stackoverflow.com/questions/34548648/laravel-relationship-with-a-pivot-table) | I have 3 tables:  Category ( id, name)  Category\_Tournament (category\_id, tournament\_id) --> pivot table Category\_Tournament\_User (category\_id, tournament\_id, user\_id, confirmed)  Category is the list of available categories Category\_Tournament is the list of categories the admin configured Category\_tournament\_User is the categories the user has registred  To get all categories in the tournament, I can do it easily with:  tournament->categories  defining a belongsToMany relationship in tournament model  What I don't know how to define relationship with the last table.  What I need is the user click on several categories, and I can run something like:  tournament->user\_categories->sync($userCategories)  where I should sync table Category\_Tournament\_User ( with category\_id, tournament\_id, user\_id)  What is the best way to achieve it???  EDIT:  Model Tournament:  class Tournament extends Model  {  protected $table = 'tournament';  public $timestamps = true;  protected $fillable = [  'name',  'date',  'type',  ];  /\*\*  \* A tournament is owned by a user  \*  \* @return \Illuminate\Database\Eloquent\Relations\BelongsTo  \*/  public function owner()  {  return $this->belongsTo('App\User', 'user\_id','id');  }  /\*\*  \* @return \Illuminate\Database\Eloquent\Relations\HasMany  \*/  public function categories()  {  return $this->belongsToMany('App\Category')  ->withTimestamps();  }  }  Model Category  class Category extends Model  {  protected $table = 'category';  public $timestamps = true;  protected $fillable = [  'id',  'name',  ];  public function tournaments()  {  return $this->belongsToMany('App\Tournament');  }  }  Model User:  class User extends Model implements AuthenticatableContract, CanResetPasswordContract  {  use Authenticatable, Authorizable, CanResetPassword, HasRole;  /\*\*  \* The database table used by the model.  \*  \* @var string  \*/  protected $table = 'users';  /\*\*  \* The attributes that are mass assignable.  \*  \* @var array  \*/  protected $fillable = ['name','firstname','lastname','email', 'password','avatar',country\_id','role\_id',,'provider','provider\_id','verified'];  /\*\*  \* The attributes excluded from the model's JSON form.  \*  \* @var array  \*/  protected $hidden = ['password', 'remember\_token'];  /\*\*  \* Boot the model.  \*  \* @return void  \*/  public static function boot()  {  parent::boot();  static::creating(function ($user) {  $user->token = str\_random(30);  });  }  public function role()  {  return $this->belongsTo('App\Role');  }  public function settings()  {  return $this->hasOne('App\Settings');  }  public function invites()  {  return $this->hasMany('App\Invite', 'email','email');  }  public function country()  {  return $this->belongsTo('Webpatser\Countries\Countries');  }  /\*\*  \* A user can have many tournaments  \*  \* @return \Illuminate\Database\Eloquent\Relations\HasMany  \*/  public function tournaments()  {  return $this->hasMany('App\Tournament');  }  } |