☐ cmtl0204 / siga_backend

Code Issues Pull requests 1 Actions Projects Wiki Security Insights

```
ြီး d6da22285c ▼ ...
```

siga_backend / app / Models / JobBoard / Skill.php / <> Jump to ▼

```
cmtl0204 asd

8x 2 contributors  

©
```

```
90 lines (77 sloc) 2.13 KB
      <?php
      namespace App\Models\JobBoard;
      use Illuminate\Database\Eloquent\Model;
      use Illuminate\Database\Eloquent\Factories\HasFactory;
      use OwenIt\Auditing\Contracts\Auditable;
      use OwenIt\Auditing\Auditable as Auditing;
      use Illuminate\Database\Eloquent\SoftDeletes;
      use Dyrynda\Database\Support\CascadeSoftDeletes;
      use App\Models\App\File;
      use App\Models\App\Image;
      use App\Models\App\Catalogue;
       * @property BigInteger id
       * @property string description
       */
      class Skill extends Model implements Auditable
          use HasFactory;
          use Auditing;
          use SoftDeletes:
          use CascadeSoftDeletes;
          protected static $instance;
           protected $connection = 'pgsql-job-board';
           protected $table = 'job_board.skills';
           protected $with = ['professional', 'type'];
           protected $fillable = [
               'description',
```

```
1;
protected $casts = [
    'deleted at' => 'date:Y-m-d h:m:s',
    'created at' => 'date:Y-m-d h:m:s',
    'updated_at' => 'date:Y-m-d h:m:s',
protected $cascadeDeletes = ['files'];
public static function getInstance($id)
    if (is_null(static::$instance)) {
        static::$instance = new static;
    static::$instance->id = $id;
   return static::$instance;
}
// Relationships
public function professional()
{
   return $this->belongsTo(Professional::class);
}
public function type()
   return $this->belongsTo(Catalogue::class);
public function files()
    return $this->morphMany(File::class, 'fileable');
public function images()
   return $this->morphMany(Image::class, 'imageable');
}
// Accessors
public function getFullDescriptionAttribute()
   return "{$this->attributes['id']}.{$this->attributes['description']}";
// Mutators
public function setDescriptionAttribute($value)
    $this->attributes['description'] = strtoupper($value);
// Scopes
public function scopeDescription($query, $description)
```