

# Cas Music

## Présentation du contexte

---

### L'organisation cliente : l'école de musique municipale de la ville de D.

Inscrite dans le paysage culturel de la ville de D. (15 000 habitants) depuis une quarantaine d'années, l'école de musique municipale compte aujourd'hui quelques 300 usagers.

Ouverte à tous, enfants, adolescents, adultes, et couvrant le territoire de la communauté de communes, elle a pour vocation l'apprentissage de la musique.

Son projet d'établissement est axé vers les grandes missions suivantes :

- ✓ formation des musiciens (instrumentale, musicale, orchestrale, ateliers, etc.) ;
- ✓ diffusion, création (concerts, auditions) ;
- ✓ sensibilisation d'un large public et développement de partenariats locaux (éducation nationale, scène nationale voisine, maison des jeunes et de la culture, centres d'animation, pratiques amateurs, *master class*, conférences).

Dotée de locaux de qualité, l'école de musique dispose d'une équipe pédagogique de seize enseignants et propose au public différentes prestations d'enseignement musical.

L'école dispense des cours d'instruments individuels ou collectifs pour enfants, adolescents et adultes, en fonction des places disponibles. L'apprentissage est accessible à partir de 7 ans, sauf précision particulière pour certains instruments.

### L'entreprise prestataire de services

La DSI - direction des systèmes d'information de la mairie de D. - regroupe deux personnes à temps plein afin de répondre aux besoins des différentes entités de la mairie réparties sur une dizaine de sites.

L'école de musique municipale est placée sous la tutelle de la mairie et profite à ce titre de services informatiques fournis par la DSI : bureautique, messagerie, accès au réseau et à internet, maintenance préventive des postes de travail et assistance aux utilisateurs.

La DSI assure également la fourniture de services spécifiques à l'école de musique : gestion des inscriptions, facturation et suivi des règlements.

### Le projet

Avec les outils informatiques actuellement en place, la gestion de l'école de musique est jugée peu efficace par les personnels de l'école. La DSI de la mairie mène une étude en vue de moderniser ces outils.

| Sup De Vinci |  | Etude de cas |             |
|--------------|--|--------------|-------------|
| Durée : 3h   |  |              | Page : 1/13 |

Intégré-e en tant que salarié-e à la DSI, vous assistez M. Lejoyau, responsable de la DSI, et participez à cette étude et aux différentes missions qui la composent.  
Un dossier documentaire vous est remis pour compléter votre information.

## Évolution du logiciel Musicol

---

La communauté de communes alloue depuis de nombreuses années un budget annuel pour la maintenance et le renouvellement d'un parc d'instruments de musique. Ce parc est destiné à deux utilisations distinctes :

- ✓ des instruments qui restent à l'école de musique, par exemple des instruments ne pouvant facilement être déplacés (piano, harpe...) ou alors des instruments mis à disposition des professeurs pour leurs cours et ateliers ;
- ✓ des instruments qui sont prêtés gracieusement aux élèves débutants, ce qui évite aux familles d'investir, dès le début de l'apprentissage, dans un instrument souvent coûteux.

Pour le moment, la gestion du parc est manuelle et s'effectue de la façon suivante :

- ✓ Chaque instrument possède une étiquette discrètement collée sur laquelle est inscrit un code (par exemple le code "**bas-2**" signifie qu'il s'agit de la 2<sup>ème</sup> basse électrique du parc) et une fiche sur laquelle figure la description détaillée, une photo ainsi que les interventions de réglage et de réparation.
- ✓ Pour les instruments de prêt, une deuxième fiche récapitule les prêts aux élèves.

Dans le cadre de la maintenance du parc, l'école de musique travaille avec plusieurs professionnels partenaires (accordeurs de piano, luthiers, facteurs de divers instruments, ...) à qui l'on confie les instruments pour des réglages ou des réparations. Au retour de chaque intervention, la fiche de l'instrument concerné est alors complétée.

En début d'année, les familles qui désirent emprunter un instrument font une demande motivée (non stockée dans la base de données). Une commission se réunit pour attribuer les instruments, la priorité étant donnée aux débutants. Le jour de la remise de l'instrument, l'école de musique établit un contrat avec la famille qui contient les éléments suivants :

- ✓ la date de signature et la date de fin de contrat généralement située vers le 7 septembre de l'année suivante ;
- ✓ un descriptif de l'état de l'instrument qui fait suite à une vérification minutieuse effectuée en présence de la famille dans lequel chaque détail est noté (par exemple un petit enfoncement dû à un choc,...) ;
- ✓ le numéro de la police d'assurance de la famille qui couvre les risques liés à l'usage de l'instrument (casse, perte ou vol).

Il est à noter qu'un instrument de prêt est fourni avec des accessoires (housse, câble jack, archet, amplificateur...) indispensables pour sa pratique et sa protection. Lors de la restitution de l'instrument, un an plus tard, le contrat est complété par l'état global. Si l'instrument doit subir un réglage ou une réparation non due à une usure normale, l'école de musique est en droit de facturer tout ou partie du prix de l'intervention, dans ce cas on notera la quotité demandée à la famille (100% pour tout le montant, 50% pour la moitié du montant,...).

### Votre mission

- Fournir un schéma de données présentant les évolutions de la structure de la base de données relatives à la gestion du parc d'instruments : acquisitions, prêts, interventions de réglage et de réparation.**
- Réaliser une webApp avec les technologies de votre choix mais en intégrant des requêtes SQL à la bdd définie en question 1.**

| Sup De Vinci |  | Etude de cas |             |
|--------------|--|--------------|-------------|
| Durée : 3h   |  |              | Page : 2/13 |

## Ressources documentaires

### 2. Extrait du script *donneesBase.sql* (base de données du logiciel Musicol)

#### INSERT INTO instruments values

```
("', 'Batterie'),  
('', 'Piano');
```

#### INSERT INTO cours VALUES

```
("', 'batterie', 8, NULL, NULL, 1, 1),  
('', 'piano', 8, NULL, NULL, 2, 1),  
('', 'atelier jazz', 12, NULL, 10, NULL, 2);
```

#### INSERT INTO tranches VALUES

```
(1, 0), (2, 251), (3, 426), (4, 681), (5, 935), (6, 1801);
```

#### INSERT INTO coutier VALUES

```
(1, 1, 60.00), (1, 2, 96.00), (1, 3, 126.00), (1, 4, 192.00), (1, 5, 282.00), (1, 6, 330.00),  
(2, 1, 30.00), (2, 2, 48.00), (2, 3, 63.00), (2, 4, 96.00), (2, 5, 141.00), (2, 6, 165.00);
```

### 3. Extrait du script *structureBase.sql* (base de données du logiciel Musicol)

#### 1. CREATE TABLE inscriptions

```
2. (  
3. id INTEGER(11) NOT NULL AUTO_INCREMENT ,  
4. nombre_de_paiements INTEGER(11) NOT NULL ,  
5. id_eleve INTEGER(11) NOT NULL ,  
6. id_cours INTEGER(11) NOT NULL ,  
7. PRIMARY KEY (id) ,  
8. FOREIGN KEY (id_eleve) REFERENCES eleves (id) ,  
9. FOREIGN KEY (id_cours) REFERENCES cours (id)  
10.);
```

#### 11. CREATE TRIGGER trigger\_inscriptions\_insert BEFORE INSERT ON inscriptions

```
12. FOR EACH ROW  
13. BEGIN  
14. IF NEW.nombre_de_paiements < 1 or NEW.nombre_de_paiements > 3  
15. THEN  
16. signal sqlstate '16440' set message_text = 'Insertion interdite';  
17. END IF;  
18. END;
```

#### 19. CREATE TABLE cours

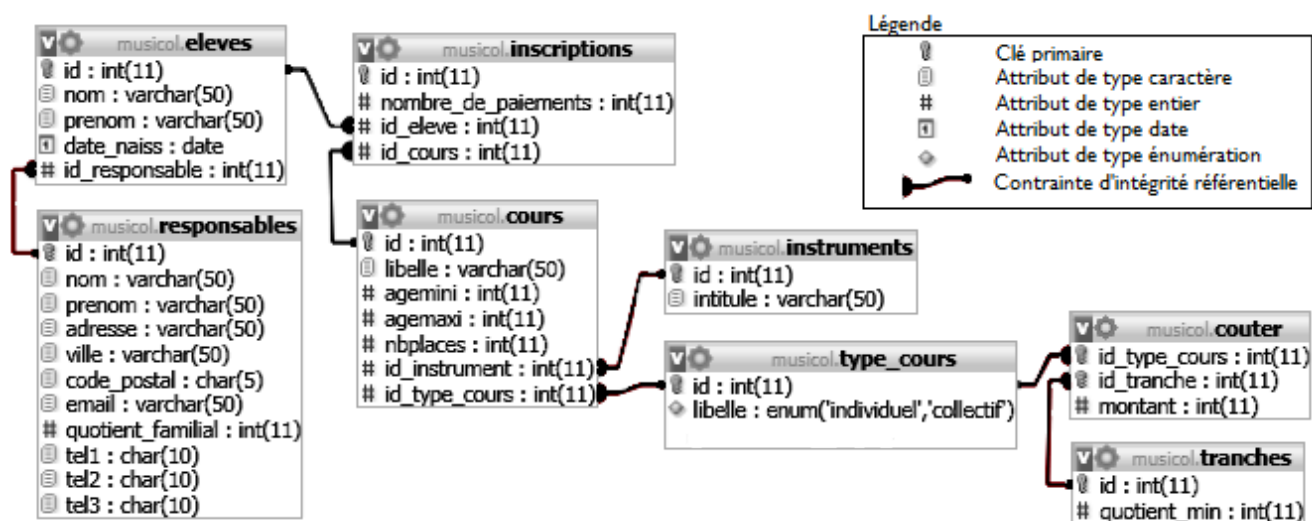
```
20. (  
21. id INTEGER NOT NULL AUTO_INCREMENT ,  
22. libelle VARCHAR(50) NOT NULL ,  
23. agemini INTEGER(11) NOT NULL ,  
24. agemaxi INTEGER(11) NULL ,  
25. nbplaces INTEGER(11) NULL ,  
26. id_instrument INTEGER(11) NULL ,  
27. id_type_cours INTEGER(11) NOT NULL ,  
28. PRIMARY KEY (id) ,  
29. FOREIGN KEY (id_instrument) REFERENCES instruments (id),  
30. FOREIGN KEY (id_type_cours) REFERENCES type_cours (id)  
31.);
```

| Sup De Vinci |  | Etude de cas |             |
|--------------|--|--------------|-------------|
| Durée : 3h   |  |              | Page : 3/13 |

#### 4. Tarifs des tranches de quotient familial

| Q.F.  | Tranche | Cours individuel | Cours collectif |
|---|---------|------------------|-----------------|
| <b><i>Résidents de la communauté de communes</i></b>      |         |                  |                 |
| 0 à 250   | A       | 60               | 30              |
| 251 à 425   | B       | 96               | 48              |
| 426 à 680   | C       | 126              | 63              |
| 681 à 934   | D       | 192              | 96              |
| 935 à 1800  | E       | 282              | 141             |
| + de 1800   | F       | 330              | 165             |
| <b><i>Résidents hors de la communauté de communes</i></b> |         |                  |                 |
|   | EXT     | 417              | 209             |

## 5. Schéma relationnel initial de la base de données Musicol



**responsables** (id, nom, prenom, adresse, ville, code\_postal, email, quotient\_familial, tel1, tel2, tel3)

Clé primaire : id

**eleves** (id, nom, prenom, date\_naiss, id\_responsable)

Clé primaire : id

Clé étrangère : id\_responsable en référence à id de responsables

**instruments** (id, intitule)

Clé primaire : id

**type\_cours** (id, libelle) // libellé a pour valeur **individuel** ou **collectif**

Clé primaire : id

**cours** (id, libelle, agemini, agemaxi, nbplaces, id\_instrument, id\_type\_cours)

Clé primaire : id

Clé étrangère : id\_instrument en référence à id de instruments

Clé étrangère : id\_type\_cours en référence à id de type\_cours

**tranches** (id, quotient\_min)

Clé primaire : id

**couter** (id\_type\_cours, id\_tranche, montant)

Clé primaire : id\_type\_cours, id\_tranche

Clé étrangère : id\_tranche en référence à id de tranches

Clé étrangère : id\_type\_cours en référence à id de type\_cours

**inscriptions** (id, nombre\_de\_paiements, id\_eleve, id\_cours)

Clé primaire : id

Clé étrangère : id\_eleve en référence à id de eleves

Clé étrangère : id\_cours en référence à id de cours

| Sup De Vinci |  | Etude de cas |             |
|--------------|--|--------------|-------------|
| Durée : 3h   |  |              | Page : 5/13 |

## 6. Écran de présentation de la grille tarifaire (produit par grilleView.php)

### Tarifs de l'année 2014

Cette indication doit être remplacée par EXT dans la nouvelle version de la page grilleView.php

| Quotient familial | Cours individuel | Cours collectif |
|-------------------|------------------|-----------------|
| -9999 à -1        | 417              | 209             |
| 0 à 250           | 60               | 30              |
| 251 à 425         | 96               | 48              |
| 426 à 680         | 126              | 63              |
| 681 à 935         | 192              | 96              |
| 936 à 1800        | 282              | 141             |
| 1801 et supérieur | 330              | 165             |

## 7. Éléments de documentation PHP

Implémentation d'une collection d'objets :

```
$maCollection = array() ; // Création de la collection
$maCollection[] = $objetUn ; // Ajout d'un objet à la collection
$maCollection[] = $objetDeux ;
$maCollection[] = $objetTrois ;
foreach($maCollection as $unObjet) // Parcours de la collection
{
    // Utilisation de l'objet $unObjet
    echo $unObjet->GetLibelle() ; // Appel d'une méthode de l'objet
}
```

Implémentation d'un dictionnaire<clé, valeur> de <chaîne, objet> :

Remarque : une particularité du langage PHP est de permettre de parcourir un dictionnaire en récupérant à la fois la clé et la valeur de chaque élément.

```
$monDico = array() ; // Création du dictionnaire, ou tableau associatif
$monDico["un"] = $unObjet1 ; // Ajout du couple <"un", unObjet1> au dictionnaire
$monDico["deux"] = $unObjet2 ; // Ajout du couple <"deux", unObjet2 > au dictionnaire
$monDico["trois"] = $unObjet3 ; // Ajout du couple <"trois", unObjet3 > au dictionnaire
$unObjet = $monDico["deux"] ; // $unObjet est affecté de unObjet2
if (isset($monDico["deux"]))... // Ce test retourne vrai, car la clé "deux" existe
if (isset($monDico["quatre"]))... // Ce test retourne faux, la clé "quatre" n'existe pas

foreach($monDico as $laCle=>$unObjet) // Parcours du dictionnaire
{
    // Utilisation de l'objet $unObjet
    echo $laCle; // Affichage de la clé
    echo $unObjet->UneMethode(); // Appel d'une méthode de l'objet
}
```

## 8. Présentation et code de la page grilleView.php

| Sup De Vinci |  | Etude de cas |             |
|--------------|--|--------------|-------------|
| Durée : 3h   |  |              | Page : 6/13 |

La vue **grilleView.php** permet l’affichage de la grille tarifaire. Elle dispose de l’année courante et des 3 tableaux suivants :

- Le tableau associatif (ou dictionnaire) **\$lesTranches** contient l’ensemble des tranches de quotient familial : l’indice est l’identifiant de tranche ; la valeur est un objet de la classe *Tranche* exposant deux méthodes publiques pour obtenir l’une le quotient minimum de la tranche, l’autre le quotient maximum de la tranche.
- Le tableau **\$lesTarifs** contient l’ensemble des tarifs de l’école, présentant deux dimensions : la première est l’identifiant du type de cours, la seconde est l’identifiant de la tranche. La valeur contenue est le tarif associé. Par exemple : *\$lesTarifs[1][2]* contient le tarif du type de cours 1 pour la tranche 2 (quotient compris entre 251 et 425).
- Le tableau associatif (ou dictionnaire) **\$lesTypesCours** contient l’ensemble des types de cours : l’indice est l’identifiant de typeCours ; la valeur est un objet de la classe *TypeCours*.

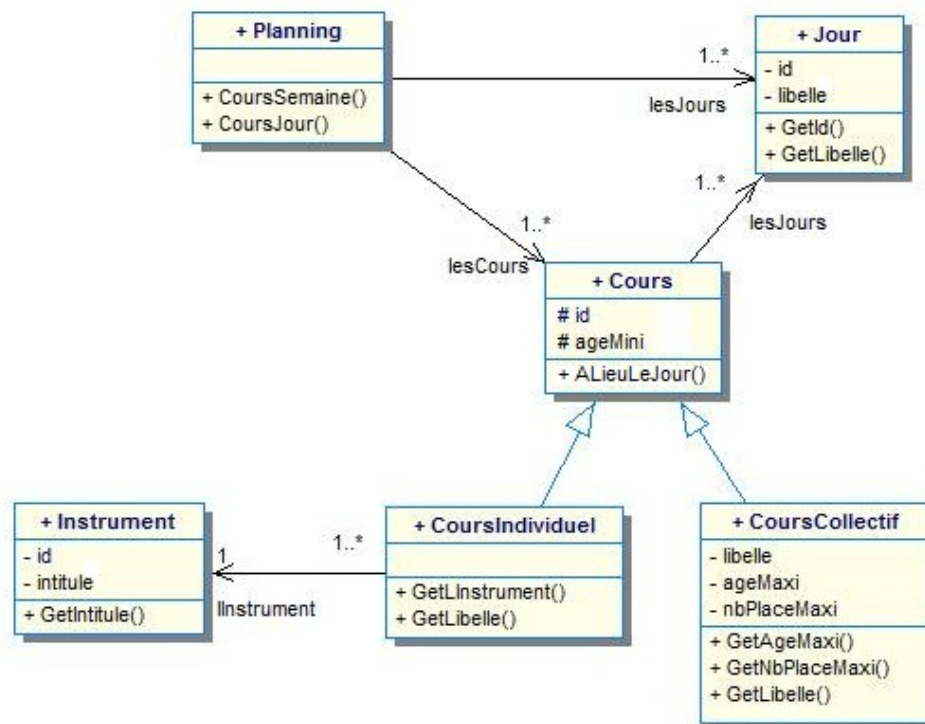
```
<div class="tarif">
<h2>Tarifs de l'année <?php echo $annee ?></h2>
<table id="grille">

    <thead>
        <tr>
            <th>Quotient familial</th>
            <th>Cours individuel</th>
            <th>Cours collectif</th>
        </tr>
    </thead>

    <?php $i=0;
    foreach($lesTranches as $tran_id=>$tranche){
        if($i % 2){
            echo "<tr class='altrow'>";
        }
        else{
            echo "<tr>";
        }
        $i++;
        echo "<td>";
        if ($i<count($lesTranches)) {
            echo $tranche->getQuotientMin() .
                " à ". $tranche->getQuotientMax();
        }
        else {
            echo $tranche->getQuotientMin() . " et supérieur";
        }
        echo "</td>";
        foreach ($lesTypesCours as $typeCours_id=>$typeCours){
            echo "<td>";
            echo $lesTarifs[$typeCours_id][$tran_id];
            echo "</td>";
        }
        echo "</tr>";
    }
?>
</table>
</div>
```

|              |  |              |             |
|--------------|--|--------------|-------------|
| Sup De Vinci |  | Etude de cas |             |
| Durée : 3h   |  |              | Page : 7/13 |

## 9. Diagramme de classes pour la planification des cours



## 10. Code PHP des classes métier pour la planification des cours (extrait)

```

<?php
class Jour{
    private $id;
    private $libelle; // libellé du jour

    public function __construct($unId, $unLibelle){ // constructeur
        $this->id=$unId;
        $this->libelle=$unLibelle;
    }

    public function GetId(){
        return $this->id;
    }

    public function GetLibelle(){
        return $this->libelle;
    }
}

class Instrument{
    private $id;
    private $intitule; // intitulé de l'instrument

    public function __construct($unId,$unIntitule){ // constructeur
        $this->id=$unId;
        $this->intitule=$unIntitule;
    }

    public function GetIntitule(){ // retourne l'intitulé de l'instrument
        return $this->intitule;
    }
}
  
```



```

    }
}
abstract class Cours{
    protected $id;
    protected $ageMini;
    protected $lesJours; // collection des jours où a lieu le cours

    public function __construct($unId,$unAgeMini){ // constructeur
        $this->id=$unId;
        $this->ageMini=$unAgeMini;
        $this->lesJours=array();
    }

    public function ALieuLeJour(Jour $unJour){    }
    //renvoie vrai si le cours a lieu le jour passé en paramètre
}

```

```

class CoursCollectif extends Cours{
    private $libelle; // libellé du cours collectif
    private $ageMaxi;
    private $nbPlacesMaxi;

    // constructeur
    public function __construct($unId,$unAgeMini,$unLibelle,$unAgeMaxi,$unNbPlacesMaxi){
        parent::__construct($unId,$unAgeMini);
        $this->libelle=$unLibelle;
        $this->ageMaxi=$unAgeMaxi;
        $this->nbPlacesMaxi=$unNbPlacesMaxi;
    }

    public function GetLibelle(){
        // retourne le libellé du cours collectif
    }

    public function GetAgeMaxi(){
        return $this->ageMaxi;
    }

    public function GetNbPlacesMaxi(){
        return $this->nbPlacesMaxi;
    }
}

```

```

class CoursIndividuel extends Cours{
    private $Instrument; //instrument concerné par le cours individuel

    public function __construct($unId,$unAgeMini,$unInstrument){ // constructeur
        parent::__construct($unId,$unAgeMini);
        $this->Instrument=$unInstrument;
    }
    public function GetInstrument(){
        return $this->Instrument ;
    }

    public function GetLibelle(){
        // retourne le libellé du cours individuel qui est en fait l'intitulé de l'instrument
    }
}

```

**class Planning**{

| Ynov |  | Etude de cas |             |
|------|--|--------------|-------------|
|      |  |              | Page : 9/13 |

```

public $lesJours; // collection des jours de la semaine
public $lesCours; // collection des cours

public function __construct(){ //constructeur
    $this->lesJours=array();
    $this->lesCours=array();
}

public function CoursJour(Jour $unJour){
    // retourne une collection d'objets Cours contenant les cours du jour $unJour
}

public function CoursSemaine(){
    $lePlanningComplet=array();
    foreach($this->lesJours as $leJour){
        $lePlanningComplet[$leJour->GetLibelle()] =$this->CoursJour($leJour);
    }
    return $lePlanningComplet;
}
?>

```

## 11. Fiche de renseignement d'un instrument

**bas-2** (Basse électrique)  
 Date d'achat : 06/06/2011  
 Prix d'achat : 380 €  
 Marque : Squier Modèle : Jazz Bass  
 Numéro de série : Q 07219  
 Couleur dominante : Noire  
 Utilisation : prêt



**Classe d'instruments** : Instruments amplifiés

### Interventions

| N° | Date début | Date fin   | Professionnel  | Travaux effectués              | Prix |
|----|------------|------------|--|--------------------------------|------|
| 1  | 12/09/2013 | 14/09/2013 | Hall de musique<br>24 rue Amiral Ronarc'h<br>29000 Quimper | Réglage du manche              | 30€  |
| 2  | 12/09/2014 | 12/09/2014 | Ets Poënot<br>Port Rhu<br>29100 Douarnenez                 | Réparation de<br>la prise jack | 10€  |
| 3  | 14/09/2014 | 15/09/2014 | Hall de musique<br>24 rue Amiral Ronarc'h<br>29000 Quimper | Réglage du manche              | 32€  |

## 12. Fiche de prêts d'un instrument

### bas-2 (Basse électrique)

Accessoires fournis : housse souple, jack, ampli 10W



#### Prêts

| Elève   | Date début | Date fin   | N° intervention<br>suite à un prêt | quotité<br>famille |
|---|------------|------------|------------------------------------|--------------------|
| Horellou Simon<br>15, rue Ar Ster<br>29100 Pouldergat         | 14/09/2011 | 07/09/2012 |                                    |                    |
| Queffelec Annaël<br>31 rue Édouard Branly<br>29100 Douarnenez | 15/09/2012 | 08/09/2013 | 1                                  | 0%                 |
| Le Boulch Thibault<br>56 rue Victor Hugo<br>29100 Douarnenez  | 16/09/2013 | 09/09/2014 | 2<br>3                             | 100%<br>0%         |

**Remarque** : Le prêt à l'élève Simon Horellou n'a nécessité aucune intervention.

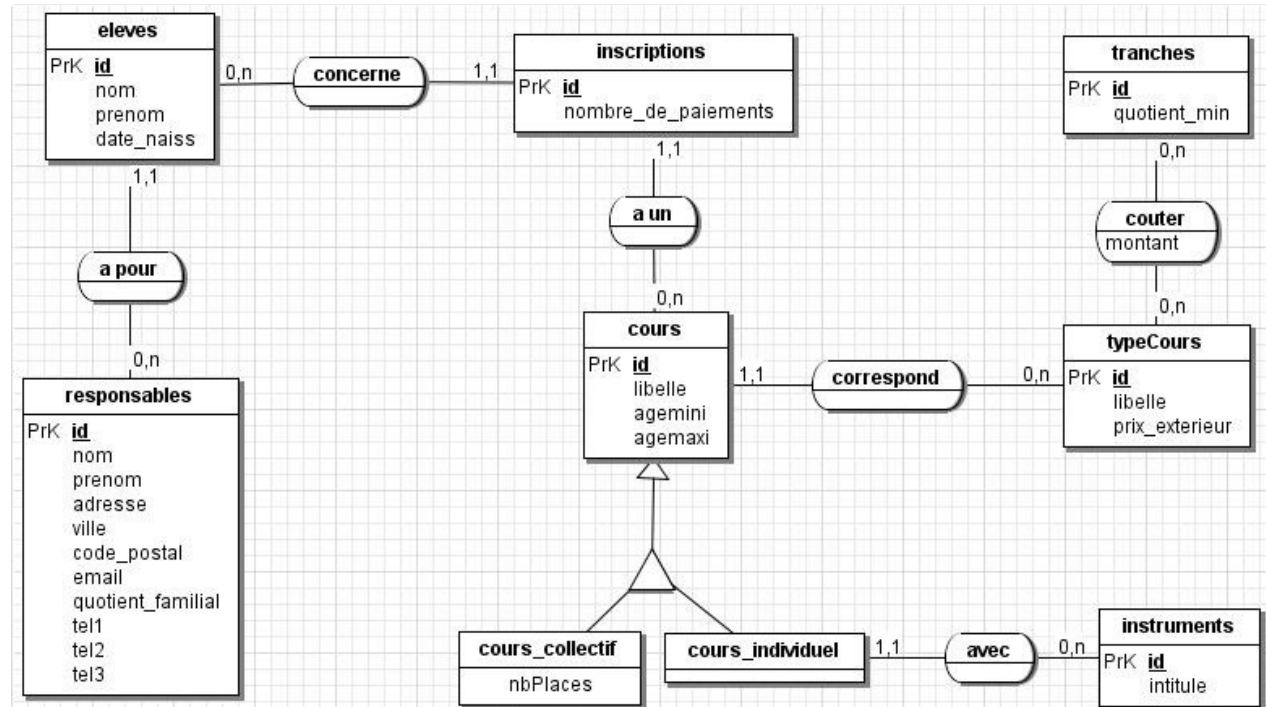
## 13. Liste des instruments enseignés et leur classe

| Classes d'instruments    | Instruments        |
|--------------------------|--------------------|
| Claviers                 | Orgue              |
|                          | Piano              |
| Instruments<br>amplifiés | Clavier amplifié   |
|                          | Guitare électrique |
|                          | Basse électrique   |
| Bois                     | Saxophone          |
|                          | Clarinette         |
|                          | Flute traversière  |
| Cuivres                  | Trombone           |
|                          | Trompette          |
|                          | Tuba               |
| Cordes                   | Violon             |
|                          | Violoncelle        |
|                          | Harpe celtique     |
| Percussions              | Batterie           |

## 14. Modélisation conceptuelle de la base de données modifiée Musicol

*La modélisation conceptuelle est volontairement fournie sous les deux représentations les plus courantes.*

### Schéma entité-association



## Diagramme de classes

