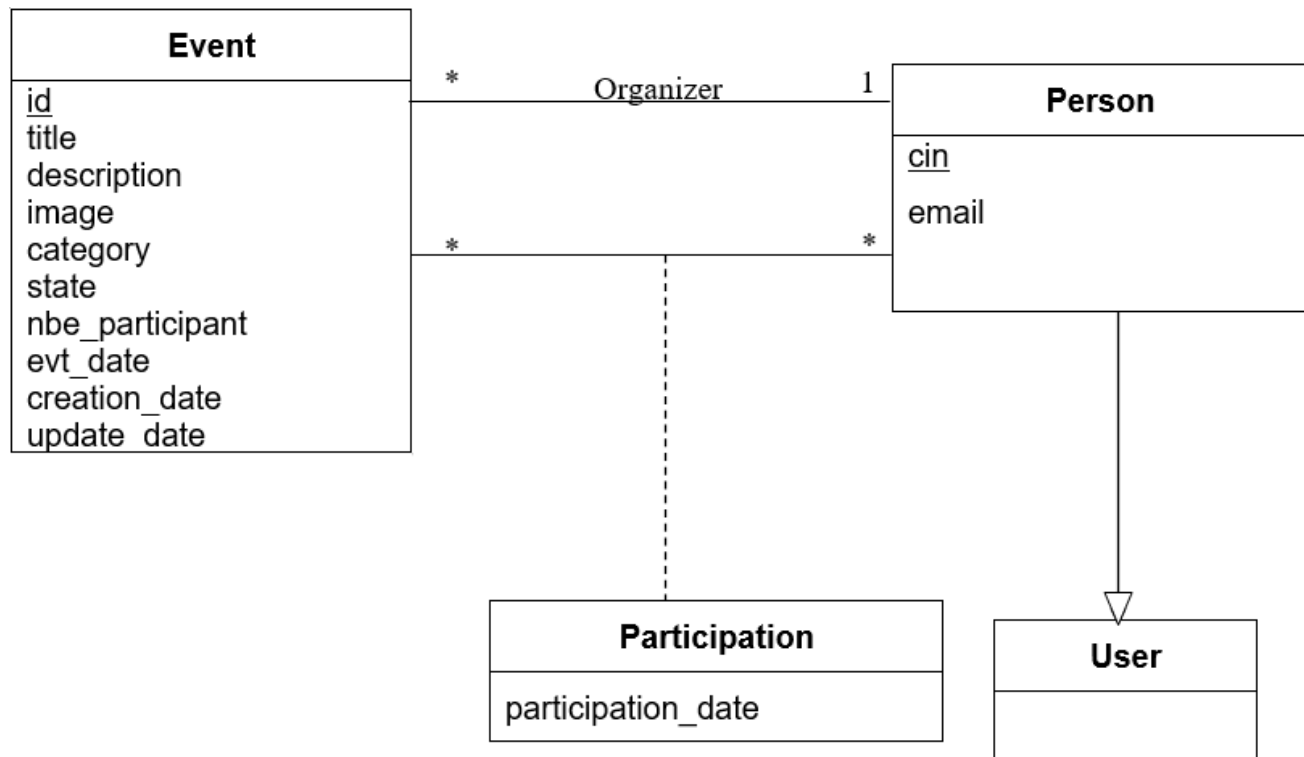


# Etude de cas: Génération de la BD

---

# Enoncé

L'objectif de cet atelier est de développer une application web de gestion des événements en utilisant le Framework Django pour les deux parties FrontOffice et BackOffice. Cette application est représentée par le diagramme de classe d'analyse suivant :



**Figure:** Diagramme de classe d'analyse

## I. Mise en place du projet

Pour la mise en place du projet on besoin:

- Installer Python 3.11
- Installer Django 4.2
- Créer un environnement virtuel nommé « **venv** »
- Créer un projet nommé « **GestionEvenement** »
- Créer deux applications : **Event** et **Person**

## II. Génération de la BD

1.Développer les entités dans le fichier « models.py » en se basant sur le diagramme de classe ci-dessus et en appliquant les contraintes suivantes :

=>Le modèle « **Person** » hérite de l'entité « **User** » prédéfini. Il faut changer la clé primaire « id » à « cin » qui doit avoir une longueur fixe égale à 8.

=>Le modèle « **Event** » doit respecter les contraintes suivantes :

\***Description** doit être de type « TextField ».

\***Image** doit être de type « ImageField », il faut installer la librairie **Pillow** via cette commande :

```
pip install Pillow
```

\***State** (bool, default=False)

\***Email** doit se terminer par "@esprit.tn".

\***Category** doit représenter une liste de choix comme suit:

-Musique

-Cinema

-Sport

\*Un événement est organisé par une seule personne et une personne peut organiser plusieurs événements.

\*Une personne peut participer à plusieurs événements. Un événement peut avoir plusieurs participants.

\*Déclarer la classe **Meta** dans laquelle on définit la contrainte sur l'attribut « **evt\_date** » à date now» .

=>Le modèle « **Participation** » est comme suit :

\*L'attribut « **participation\_date** » doit avoir comme valeur par défaut la **date système**.

\*Une personne ne peut pas participer plusieurs fois au même événement.

2.Modifier le fichier **settings.py** pour que le nom de la BD porte le nom de la classe dont vous faites partie.

3. Générer les fichiers de migration via la commande

```
python manage.py makemigrations NomApp
```

4.Générer le schéma de la BD via la commande

```
python manage.py migrate
```